

AMSTRAD

REVISTA INDEPENDIENTE PARA USUARIOS DE ORDENADORES AMSTRAD

Semanal

AÑO II N.º 61

160 Ptas.

Canarias 165 pts.

GESTOR
MULTIWINDOW
PARA DEPURACION
DE PROGRAMAS.

Para...
PCW

GESTIONA
TUS RECIBOS CON
NUESTRO PAQUETE
PROFESIONAL

Así se hace
una base
de datos
en 4 minutos
con DBASEII

MOLECULAS DE
SILICIO Y...
LOGO

LA VELOCIDAD
DEL AZAR EN
CODIGO MAQUINA



C-10 Convertidor de monitor en TV

SINTONIZA.



Preparado para todos aquellos monitores con entrada RGB LINEAL o video compuesto: AMSTRAD, COMMODORE, PHILIPS, HANTAREX, etc.

Convierte cualquier monitor en color con entrada RGB-LINEAL o PAL en una T.V. color de alta calidad de imagen. De un manejo muy sencillo, no es necesario efectuar ninguna modificación en el monitor. Su uso no produce deterioro ni alteración alguna en el funcionamiento del monitor y su diseño le hace perfectamente acoplable debajo del mismo.

ESPECIFICACIONES:

- 3 bandas
- Presintonía de 8 canales
- Salida RGB-LINEAL
- Entrada y salida de video
- Entrada y salida de audio
- Amplificador de sonido y altavoz incorporados

**conectamos
con tus ideas**

MHT ingenieros



DISTRIBUIDO POR LSB, S.A. C/. SANCHEZ PACHECO, 78. 28002 MADRID. TEL. 413 92 68

Director Editorial
José I. Gómez-Centurión

Director Ejecutivo
José M.^a Díaz

Redactor Jefe
Juan José Martínez

Diseño gráfico
José Flores

Colaboradores

Eduardo Ruiz
Javier Barceló
David Sopuerta
Robert Chatwin
Francisca Partolo
Pedro Sudón
Miguel Sepúlveda
Francisca Martín
Jesús Alonso
Pedra S. Pérez
Amalia Gámez
Alberto Suñer

Secretaría Redacción
Carmen Elías

Fotografía
Carlos Candel
Chema Sacristán

Portada

Angel Luis González

Ilustradores

J. Igual, J. Pons, F. L. Frontan,
J. Septien, Pejo, J. J. Mora

Edita

HOBBY PRESS, S.A.

Presidente

Maria Andino

Consejero Delegado
José I. Gómez-Centurión

Jefe de Producción
Carlos Peropadre

Marketing

Marta García

Jefe de Publicidad
Concha Gutiérrez

Secretaría de Dirección
Pilar Arestizábal

Suscripciones

M.^a Rosa González
M.^a del Mar Calzada

**Redacción, Administración
y Publicidad**

Ctra. de Irún km 12,400
(Fuencarral) 28049 Madrid

Pedidos y suscripciones:
734 65 00

Redacción: 734 70 12

Dto. Circulación

Paulino Blanco

Distribución

Coedis, S. A. Valencia, 245
Barcelona

Imprime

ROTEDEC, S. A. Crto. de Irún,
Km. 12,450 (MADRID)

Fotocomposición

Navocomp, S.A.
Nicolás Marales, 38-40

Fotomecánica

GROF

Ezequiel Salana, 16

Depósito Legal:
M-28468 1985

Derechos exclusivos
de la revista

**COMPUTING with
the AMSTRAD**

Representante para Argentina, Chile,
Uruguay y Paraguay, Cio.
Americana de Ediciones, S.R.L. Sud
América 1.532. Tel.: 21 24 64. 1209
BUENOS AIRES (Argentina).

M. H. AMSTRAD no se hace
necesariamente solidaria de las
opiniones vertidas por sus
colaboradores en los artículos
firmados. Reservados todos los
derechos.

MICROHOBBY

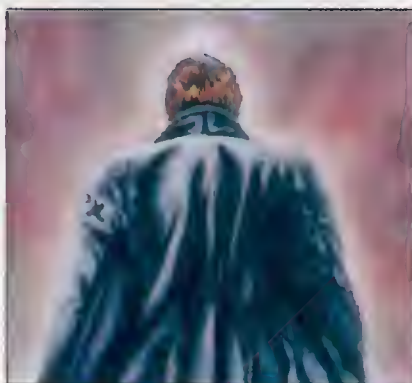
AMSTRAD

Sumario

Año II • Número 61 • 11 al 17
de Noviembre de 1986
160 ptas. (incluido I.V.A.)
Canarias, 155 ptas. + 10 ptas. sobretasa aérea
Ceuta y Melilla, 155 ptas.

Primera 5 Plana

Das páginas de noticias con toda la actualidad
informática, especialmente de **AMSTRAD**.



8 Código máquina

Durante das semanas hemos abierto el **AMSTRAD**
para sacar números aleatorios. Con este artículo
ponemos el broche final al tema con algunas sencillas
pero espectaculares rutinas que estamos seguras, serán
del aprecio de todos.

Caratulas 12 de disco

Publicamos nuevamente una serie de caratulas para
tus discos. Esperamos que as gusten y saquéis el
máximo partido de ellas.



17 Gestión PCW

Un gran programa que ofrece a todas los usuarios
del PCW la posibilidad de gestionar sus recibos. Como
por desgracia la regla «las programas buenos son
largos», vuelve a cumplirse y en esta ocasión nos
vemos obligados a darlo en dos partes.



Para... PCW 22

Quizás el programa más patente que existe hoy en
día dentro del mundo profesional para ordenadores
baja CPM sea el ya clásica DBASEII. En España existe
versión de este, tanta para el CPC 6128 como para el
PCW. Por desgracia DBASEII es también un programa
bastante difícil de utilizar, sin embargo con la lectura
de este artículo podrás utilizarlo para diseñar tus
propias bases, una de las mayores posibilidades de
este gran programa.

Análogo 28

Detenemos por una semana nuestra andadura por el
campo de la inteligencia artificial para dedicarnos a
otro tema donde el LOGO está también demostrando
su verdadera talla: la educación.

DISCOS DE CRISTAL

La firma japonesa Hoya acaba de finalizar su proyecto de un disco con soporte de vidrio.

Este tipo de disquettes dispone de una capacidad entre 5 y 10 veces superior a sus homólogos basados en soporte de aluminio. Sin embargo, lo mejor se encuentra en que su precio es similar a éstos.

El nuevo disquette ha sido desarrollado conjuntamente con empresas norteamericanas (!!!). Pero su realización se ha aplicado un material magnético sobre una superficie de vidrio.

Por ahora, el resto sigue siendo una incógnita, un secreto que parece ser, por el momento, nadie va a desvelar. Lo que sí parece estar bien claro, es que este producto no verá la luz en el mercado mientras no se haya comprobado totalmente su calidad. Habrá que esperar...

PAQUETE INTEGRADO DE GESTION DE EMPRESA DE VALLES INFORMATICA



La firma catalana **Vallés Informática, S.A.**, presenta un conjunto de dos programas con ficheros interactivos para la gestión completa de empresa. Estos son:

GESCOVISA: Paquete de gestión comercial, integrado con la contabilidad que contempla entrada de albaranes, facturas, recibos, diario de ventas, listado de IVA, control de stocks, stock bajo mínimos, inventario, comisiones por representantes y otros listados estadísticos. El fichero de clientes enlaza directamente con la contabilidad y da el alta de forma automática. Capacidad para 1.200 clientes y 2.500 artículos, con la posibilidad de tener menos clientes y más artículos, o viceversa. (Un fichero puede crecer a costa de otro).

CONTAVISA: Contabilidad, adaptada al Plan General Contable. Movimientos, diario, Balance de Sumas y Saldos, Explotación y Situación, entrada de facturas, tanto de proveedores como de clientes (Para quien no utilice la facturación) listados de IVA, etc. Capacidad para 450 cuentas y 10.000 apuntes (Disco de 700 K).

Ambas utilidades están disponibles tanto para **Amstrad CPC, PCW y PC COMPATIBLE**.

PRESENTACION AMSTRAD PCW 8256 EN LAS JORNADAS SOBRE LITERATURA E INFORMATICA EN BARCELONA



El pasado día 20 tuvieron lugar en Barcelona las **Primeras Jornadas sobre Literatura e Informática** promovidas por el Centre Divulgador de la Informática de la Generalitat de Catalunya, donde se dieron cita las más prestigiosas marcas del sector presentando el mejor software en tratamiento de textos del mercado.

Estas jornadas estaban dirigidas a

los escritores, editores y personas relacionadas con el mundo de la literatura, a fin y efecto de mejorar las condiciones de trabajo.

ACE, delegación Amstrad en Cataluña, presentó el ordenador **PCW 8256** especializado en tratamiento de textos. Paralelamente se realizó, como primicia en este país, la comunicación directa de dos ordenadores **Amstrad CPC 6128** vía telefónica (MODEM), efectuándose también una transmisión de ficheros.

Queremos resaltar el enorme éxito y favorable acogida que nuevamente obtuvieron los productos que componen la familia **Amstrad**.

PROGRAMA PARA AMSTRAD: «GUILLEM DE BERGUEDA»



Como resultado de la magnífica cooperación llevada a cabo entre el equipo de programación de ACE Software y el Centre Divulgador de la Informàtica de la Generalitat de Catalunya, nace Guillem de Berguedà versión **Amstrad** 464/6128. Si bien el programa en sí no presenta ninguna novedad, ya que el propio Centre Divulgador sigue realizando su difusión para todos los ordenadores presentes en el mercado español, la versión realizada en el ordenador **Amstrad** lo dota de unas características excepcionales, al verse enriquecido con todas las posibilidades gráficas y sonoras del ordenador.

Cuando hablamos de Guillem de Berguedà versión **Amstrad**, casi estamos hablando de un desarrollo distinto, aunque conservando, naturalmente, el magnífico guión que da vida a nuestro personaje a través de las tierras catalanas.

El argumento, como el de cualquier buena aventura gráfica, nos traslada a los tiempos de Guillem de Berguedà, trovador que vivió en el turbulento siglo XII, y fue conocido en su época por sus canciones y poesías, sus andanzas y sus aventuras amorosas.

El programa nos hace vivir con Guillem innumerables peripecias, y participar de sus esfuerzos por encontrar el cofre maravilloso donde está guardado el secreto de su destino.

Para jugar hay que hablar con la historia, y decir las acciones que queremos realizar: IR AL NORTE, COJO ESTO, DEJO ANILLO, etc. La historia se encargará de decirnos las palabras que no entiende, las cosas extrañas que hacemos o, simplemente, nos dejará por inútiles. Con buena suerte y paciencia, al final lograremos encontrar el cofre maravilloso.

Guillem de Berguedà se presenta sólo en versión cinta para los ordenadores **Amstrad** 464/472/664/6128.

LA COMPATIBILIDAD HACE LA FUERZA

Ocho fabricantes de informática europeos han firmado un acuerdo para compatibilizar sus equipos. Entre estos ocho firmantes, son de destacar por su relevancia en el mercado de la informática, Siemens, Olivetti, ICL, Bull y Thomson.

El proyecto comprende una total compatibilidad entre los distintos equipos de esta compañía. La iniciativa puede considerarse como uno de los intentos más serios por derrocar al gran líder del otro lado del Atlántico.

Primera PLANA

LANZAMIENTO DEL DESKPRO 386 DE COMPAQ

Compaq acaba de sacar al mercado el Deskpro, el primer compatible PC, que integra en su seno el procesador 80386 de Intel.

La memoria de este diablo es de un mega de base, aunque puede ser ampliada, sin paginación, hasta 8 megas; junto a esta brutal memoria se incorporará también una unidad de disco de 1.200 K.

La frecuencia de reloj de Deskpro es de 16 megaherzios, basta decir que el PC de IBM corre a 4,77 megaherzios.

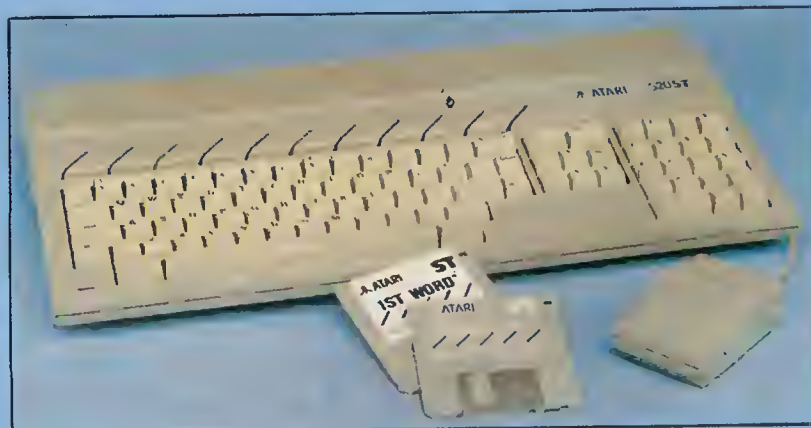
Un único problema el precio, alrededor del millón, y esto al cambio, claro está.

ATARI ST: UN MUNDO EN TRES DIMENSIONES

La empresa norteamericana Atari ha lanzado al mercado un nuevo producto en forma de libro. Se trata de Graphisme en 3 D. Un volumen en el que se realiza un exhaustivo estudio de las posibilidades gráficas de su último producto el Atari ST.

La lectura de este libro permitirá adquirir un conocimiento global sobre la programación encaminada al grafismo en tres dimensiones, así como utilizar complicadas técnicas de animación.

En la última parte de la obra se recogen numerosas rutinas explicadas a fondo, que permiten comenzar en este fabuloso campo de la informática.





FEE

INTERN

AMSTRAD

¡Ven a conocer el apasionante mundo de

Un mundo que comienza con el espectacular SPECTRUM+2 y se cierra con el revolucionario AMSTRAD PC 1512, pasando por toda la gama de los increíbles CPC 464, CPC 6128 y los procesadores de textos PCW 8256 y PCW 8512. Desde el ordenador de iniciación a la informática al más completo equipo profesional, reunidos en una ocasión única y singular: la Feria Internacional AMSTRAD-SINCLAIR.

Las más importantes empresas europeas se dan cita en Madrid para presentar sus más recientes productos:

Programas de acción, juego, aventuras... Programas educativos, de utilidades, lenguajes... Programas de gestión y profesionales...

Periféricos, ampliaciones de memoria, tabletas gráficas, digitalizadores, impresoras, redes de comunicación, sintetizadores de voz, correo electrónico, proceso de textos, tratamiento de imagen y gráficos...

Libros, revistas, cursos...

Una oportunidad extraordinaria para "estar al día".

¡¡Ven de compras a la Feria Internacional AMSTRAD-SINCLAIR!!

RIA

ACIONAL

sinclair

los ordenadores AMSTRAD y SINCLAIR!



- Patrocinada y organizada por AMSTRAD ESPAÑA.
- Horario continuo de 10,00 a 19,30
- Entrada: 250 Ptas.
- Sorteo de Ordenadores AMSTRAD y SINCLAIR ante los visitantes.

12,13 y 14 de Diciembre
Palacio de Exposiciones y Congresos
P.º Castellana. 99. 28046 MADRID

ESTA ES TU FERIA

BUSQUEDA DE CADENAS Y CARACTERES

Esta semana abordaremos el tema de la búsqueda de cadenas, de números o caracteres, en la memoria de nuestro ordenador.



Para ello hemos preparado dos programas, el primero de los cuales se encarga de buscar un valor, que puede ser un número o bien el código ASCII de un carácter, y el segundo realizará la función de búsqueda de una cadena de caracteres.

Cuando trabajamos en Basic nos podemos proponer la comparación de dos variables distintas, ahora bien, cuando se trabaja en código máquina, únicamente podemos realizar las comparaciones de cierto valor con direcciones de memoria, o bien con registros.

En este caso vamos a efectuar una búsqueda a través de la memoria comparando cada una de las direcciones con un valor dado de antemano.

Descripción del primer programa y sus variables

El programa número uno realiza una búsqueda sencilla, es decir, únicamente debe comparar si el valor dado es igual a la dirección de memoria actual; si no lo es, incrementamos el puntero y se pasa a la comparación de la siguiente dirección hasta encontrar dicho valor o bien hasta llegar a la última dirección de memoria que se desea chequear.

Así pues, desde Basic podemos realizar esta tarea a través del comando 'IF...THEN'.

Las variables que utiliza dicho programa las describiremos a continuación para luego poderlas comparar con las utilizadas en el programa en código máquina:

N...Dirección de memoria inicial
X...Dirección de memoria final
A...Valor a buscar

Así pues, el programa empezará a buscar a partir de la dirección de memoria inicial, y terminará cuando encuentre una dirección de memoria que contenga el mismo valor que la variable 'A', o cuando se llegue a la dirección final indicada por 'X'.

La rutina en lenguaje máquina está realizada con la ayuda del programa que vimos en un capítulo anterior, encargaba de pintar números decimales. Esto nos servirá para indicar la dirección en que se encuentra el valor dado, si existe.

Para buscar el número, utilizamos una instrucción muy útil para este caso, se trata de:

CPIR

la cual necesita ciertos datos de entrada colocados en los registros siguientes:

HL...Dirección inicial

BC...Longitud de memoria a comparar

A...Valor a comparar

De esta forma, la anterior instrucción actuará comparando el contenido del registro doble HL con el contenido del acumulador, y si esta comparación resulta falsa, incrementará el registro par HL y decrementará el registro BC y volverá a ejecutarse.

Si la comparación resulta verdadera, entonces pondrá el flag de cero a 1 y terminará la instrucción.

Así pues, el programa se escribirá de la siguiente forma:

LD HL,20000; Dirección inicial

LD BC,1000; Longitud

LD A,1; Valor a comparar

CPIR; Se efectúa la comparación
JR NZ,SIHAY; Ha encontrado un valor igual

Ejecutando dicho programa en Basic y código máquina, podremos observar la enorme diferencia de velocidad entre los dos lenguajes, ya que mientras el primero tarda varios segundos, el escrito en lenguaje máquina no nos permite darnos cuenta del tiempo invertido.

Diagrama de flujo del primer programa

Para dar par vista este primer programa, veamos cuál sería su diagrama de flujo:



Descripción del segundo programa y sus variables

El segundo programa que hemos preparado funciona de una forma similar al anterior, pero en este caso se trata de encontrar no un valor, sino varios, es decir, una cadena, que en este caso es de caracteres.

En el programa en Basic se utilizan las siguientes variables:

A\$...Valor de la cadena
L...Longitud de la cadena
B...Contador
X...Dirección final
N...Dirección inicial

En este caso, se intentará buscar un valor en memoria igual al primer valor de la cadena. En el momento en que se encuentre, se incrementará el contador '**B**' y se comparará la siguiente dirección de memoria con el segundo valor de la cadena. Si la comparación es verdadera se valvra a incrementar '**B**', de lo contrario éste se inicializará.

Esta operación se realizará hasta que los valores de las variables '**B**' y '**L**' sean iguales, lo cual indicará que se ha encontrado la cadena buscada.

En el caso del programa en código máquina, se realiza una operación exactamente igual a la descrita, pero en este caso se utilizarán las siguientes variables:

VARIABLE '**CADENV**'...Valor de la cadena
 VARIABLE '**LONCV**'...Longitud de la cadena
 REGISTRO '**B**'...Contador

Código MAQUINA

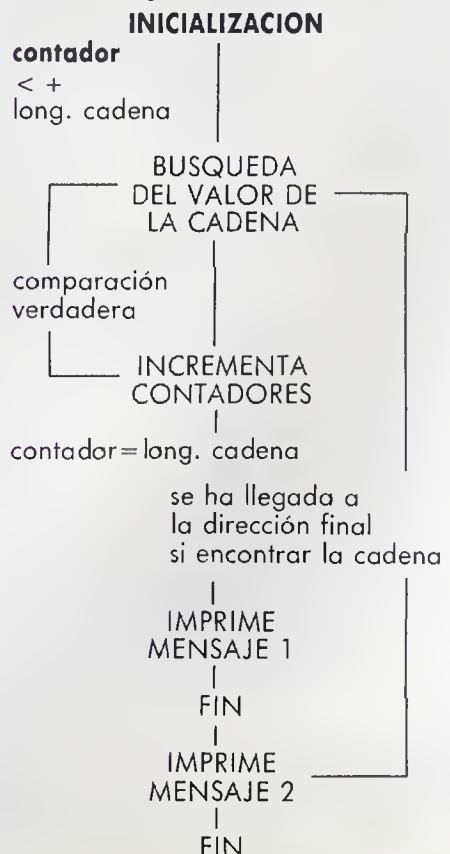
VARIABLE '**TOPE**'...Dirección final
 REGISTRO DE...Dirección inicial

En este caso, deberemos comparar el contenido de la dirección indicada por DE con el contenido de la dirección indicada por HL. Esta última apunta al contenido de la variable '**CADENV**'. En caso de comparación verdadera se incrementan **DE**, **HL** y el contador **B**, y se procede a la siguiente comparación: si ésta vuelve a ser verdadera se repite el proceso, de lo contrario se actualizan los registros **B** y **HL** y se vuelve a iniciar toda el asunto.

De esta forma el programa concluirá cuando se haya llegado a la última dirección a chequear, o bien en el momento en que se encuentre una cadena semejante a la dada en memoria.

Diagrama de flujo del segundo programa

Así pues, la descripción gráfica de este segundo programa vendría dada de la siguiente forma:



LISTADO DESENSAMBLADO 1

```

A000      10      ORG #A000
20      JHL - DIRECCION INICIAL
40      JOC - NUMERO DE BYTES A COMPARAR
50      JA - NUMERO QUE SE COMPARA
60      RET
A000 3E01      70      LD A,1
A002 C0E0BC    80      CALL #C0E0
A005 21204E    90      LD HL,20000
A006 91E0B3    100     LD BC,1000
A008 3E01      110     LD A,1
A000 ED01      120     CRIR
A00F 2007      130     JR Z,61HAY
A011 212F40    140     LD HL,TXT1
A014 C025A0    150     CALL PRINT
A017 C9        160     RET
A018 E3        170     SIHAY: PUSH HL
A019 215A00    180     LD HL,TXT2
A01C C025A0    190     CALL PRINT
A01F E1        200     POP HL
A020 20        210     DEC HL
A021 C07FA0    220     CALL PINUM
A024 C9        230     RET
A025 7E        240     PRINT: LD A,(HL)
A026 FEFF      250     CP 255
A028 C8        260     RET Z
A029 C05AB8    270     CALL #B85A
A02C 23        280     INC HL
A02D 1BF6      290     JR PRINT
A02F 44494340 310     JXT1: DEFM "DICHU VALOR NO
                    SE ENCUENTRA EN MEMORIA"
A035 FF        320     DEFB 255
A036 44494340 330     JXT2: DEFM "DICHU VALOR SE ENCUENTRA
                    EN LA DIRECCION"
A07E FF        340     DEFB 255
A07F 37        350     PINUM: SCF
A080 111027    360     LD DE,10000
A083 23        370     INC HL
A084 3E2F      380     LD A,47
A086 3C        390     INC A
A087 E052      400     SBC HL,DE
A088 30F8      410     JR NC,DMIL
A089 C0B4A8    420     CALL IMPRI
A08E 11E0B3    430     LD DE,1000
A091 3C        440     INC A
A092 ED52      450     SBC HL,DE
A094 30F8      460     JR NC,MIL
A096 C0B4A8    470     CALL IMPRI
A099 116400    480     LD DE,100
A09C 3C        490     INC A
A09D ED52      500     SBC HL,DE
A09F 30F8      510     JR NC,C1EN
A0A1 C0B4A8    520     CALL IMPRI
A0A4 118A00    530     LD DE,10
A0A7 3C        540     INC A
A0A8 ED52      550     SBC HL,DE
A0AA 30F8      560     JR NC,DIEZ
A0AC C0B4A8    570     CALL IMPRI
A0AF 05        580     ADD A,L
A0B0 C0B4A8    590     CALL IMPRI
A0B3 C9        600     RET
A0B4 C05AB8    610     IMPRI: CALL #B85A
A0B7 3E2F      620     LD A,47
A0B9 2001      630     JR NZ,PAS
A0BB 23        640     INC HL
A0BC 19        650     PAS: ADD HL,DE
A0BD 23        660     INC HL
A0BE C9        670     RET

```

LISTADO DESENSAMBLADO 2

```

A000      10      ORG #A000
A003 1160FF    20      LD DE,49800
A005 ED523EAB 30      LD (TOPE),DE
A007 11CBAF    40      LD DE,45000
A00A 213AA0    50      LD HL,CADENJ
A00D 0400      60      VULV: LD 0,0
A00F 3A3DA0    70      BUC: LD A,(LONCU)
A012 00        80      CP 0
A013 2920      90      JR Z,SIHAY
A015 E3       100     PUSH HL
A016 2A3EAB   110     LD HL,(TOPE)
A019 7A       120     LD A,D
A01A 0C       130     CP H
A01B 2004     140     JR NZ,SIGUE
A01D 70       150     LD A,E
A01E 80       160     CP L
A01F 2B11     170     JR Z,ACABA
A021 E1       180     SIDUE: PDP HL
A022 1A       190     LD A,(DE)
A023 0E       200     CP (HL)
A024 13       210     INC DE
A025 2007     220     JR Z,INCCD
A027 0400     230     LD B,0
A029 213AA0   240     LD HL,CADENJ
A02C 1BE1     250     JR BUC
A02E 04       260     INC B
A02F 23       270     INC HL
A030 180D     280     JR BUC
A032 215FA0   290     ACABA: LD HL,TXT1
A035 C055A8   300     CALL PRINT
A038 E1       310     POP HL

```

```

A039 C9       320     RET
A03A 32554E   330     CADENJ: DEFM "RUN"
A03D 03       340     LONCU: DEFB 3
A03E 00       350     TOPE: DEFB 2
A040 2107A0   360     SIHAY: LD HL,TXT2
A043 C055A8   370     CALL PRINT
A046 0B       380     EX DE,HL
A047 3A3DA0   390     LD A,(LONCU)
A04A 1600     400     LD D,0
A04C 3F       410     LD E,A
A04D 37       420     SCF
A04E 3F       430     CCF
A04F ED52     440     SBC HL,DE
A051 C0B0A0   450     CALL PINUM
A054 C9       460     RET
A055 7E       470     PRINT: LD A,(HL)
A056 FEFF     480     CP 255
A059 C0       490     RET Z
A059 C05AB8   500     CALL #B85A
A05C 23       510     INC HL
A05D 10F6     520     JR PRINT
A05F 44494340 530     JXT1: DEFM "DICHU CADENA NO
                    SE ENCUENTRA EN MEMORIA"
A064 FF       550     DEFB 255
A067 4C412043 560     JXT2: DEFM "LA CADENA SE
                    ENCUENTRA EN LA DIRECCION"

```

Hisoft DENAS.1 Assembler. Page 2.

```

A0AF FF       570     DEFB 255
A0B0 37       580     PINUM: SCF
A0B1 111027   590     LD DE,10000
A0B4 23       600     INC HL
A0B5 3E2F     610     LD A,47
A0B7 3C       620     INC A
A0B8 ED52     630     SBC HL,DE
A0BA 30F8     640     JR NC,DMIL
A0BC CDE5A0   650     CALL IMPRI
A0BF 11E0B3   660     LD DE,1000
A0C2 3C       670     MIL: INC A
A0C3 ED52     680     SBC HL,DE
A0C5 30F8     690     JR NC,MIL
A0C7 C0E5A0   700     CALL IMPRI
A0CA 116400   710     LD DE,100
A0CC 3C       720     C1EN: INC A
A0CE ED52     730     SBC HL,DE
A0D0 30F8     740     JR NC,C1EN
A0D2 C0E5A0   750     CALL IMPRI
A0D5 118A00   760     LD DE,10
A0D8 3C       770     DIEZ: INC A
A0D9 ED52     780     SBC HL,DE
A0DB 30F8     790     JR NC,DIEZ
A0DD CDE5A0   800     CALL IMPRI
A0E0 05       810     ADD A,L
A0E1 CDE5A0   820     CALL IMPRI
A0E4 C9       830     RET
A0E5 C05AB8   840     IMPRI: CALL #B85A
A0E8 3E2F     850     LD A,47
A0EA 2001     860     JR NZ,PAS
A0EC 23       870     INC HL
A0ED 19       880     PAS: ADD HL,DE
A0EE 23       890     INC HL
A0EF C9       900     RET

```

PROGRAMA BASIC 1

```

10 MODE 1
20 A=1
30 N=20000:IX=21000
40 IF PEEK(N)=A THEN PRINT "EN LA D
   DIRECCION DE MEMORIA" :N:PRINT "SE EN
   CUENTRA EL VALOR":A:END
50 IF N=N THEN PRINT "NO EXITE ESE
   VALOR EN MEMORIA":END
60 N=N+1:GOTO 40

```

PROGRAMA BASIC 2

```

10 MODE 1
20 A$="RUN":L=310:I
30 N=45000:IX=49000
40 IF PEEK(N)=ASC(MID$(A$,B,1)) THEN
   N:GOTO 70 ELSE 0=1
50 IF N=N THEN PRINT "NO EXITE ESA
   CADENA EN MEMORIA":END
60 N=N+1:GOTO 40
70 B=B+1:IF B=L THEN PRINT "DICHU
   CADENA SE ENCUENTRA ENTRE LAS " :PR
   INT "DIRECCIONES" :N-L+1:"Y":END
80 RETURN

```

PROGRAMA CARGADOR 1

```

10 REM "PROGRAMA CARGADOR"
20 FOR N=A000 TO A00F
30 READ A:SUMA=SUMA+A
40 POKE N,A
50 NEXT
60 IF SUMA<49C3 THEN PRINT "ERROR
   EN DATAS"
70 DATA 62,1,205,14,100,33,32
80 DATA 78,1,232,3,62,1,227
90 DATA 177,40,7,33,47,168,205
100 DATA 37,160,201,229,33,86,160
110 DATA 205,37,160,225,43,205,127
120 DATA 160,201,126,254,255,200,20
   5
130 DATA 90,107,35,24,246,69,73
140 DATA 47,72,79,22,06,45,76
150 DATA 79,02,32,70,79,32,03
160 DATA 69,32,69,70,67,05,69
170 DATA 70,04,02,65,32,69,70
180 DATA 32,77,69,77,79,02,73
190 DATA 65,255,68,73,67,72,79
200 DATA 32,06,65,76,79,02,32
210 DATA 03,69,32,69,70,67,05
220 DATA 69,70,04,02,65,32,69
230 DATA 70,32,76,65,32,69,73
240 DATA 02,69,67,67,73,79,70
250 DATA 255,55,17,16,39,33,62
260 DATA 47,60,237,02,40,251,205
270 DATA 100,160,17,232,3,00,237
280 DATA 02,40,251,205,100,160,17
290 DATA 100,04,237,02,40,251
300 DATA 205,100,160,17,10,0,60
310 DATA 237,02,40,251,205,100,160
320 DATA 133,205,100,160,201,205,90
330 DATA 107,62,47,32,1,35,25
340 DATA 35,201,0,0,0,0,0

```

PROGRAMA CARGADOR 2

```

10 REM "PROGRAMA CARGADOR"
20 FOR N=A000 TO A00F
30 READ A:SUMA=SUMA+A
40 POKE N,A
50 NEXT
60 IF SUMA<5003 THEN PRINT "ERROR
   EN DATAS"
70 DATA 17,104,191,237,03,62,160
80 DATA 17,200,175,33,50,160,6
90 DATA 0,50,61,160,104,40,43
100 DATA 229,42,62,160,122,100,32
110 DATA 4,123,109,40,17,225,26
120 DATA 190,19,40,7,6,0,32
130 DATA 50,160,24,225,4,35,24
140 DATA 221,33,95,160,205,05,160
150 DATA 225,201,02,05,70,3,8
160 DATA 0,33,135,160,205,05,160
170 DATA 235,50,61,160,22,0,95
180 DATA 55,42,237,02,205,176,160
190 DATA 201,124,254,255,200,205,90
200 DATA 107,35,24,246,60,73,67
210 DATA 72,65,32,67,65,60,69
220 DATA 70,65,32,70,79,32,03
230 DATA 69,32,69,70,67,05,69
240 DATA 70,04,02,65,32,69,70
250 DATA 32,77,69,77,79,02,73
260 DATA 65,255,76,65,32,67,65
270 DATA 60,69,70,65,32,03,69
280 DATA 32,69,70,67,05,69,70
290 DATA 04,02,65,32,69,70,32
300 DATA 76,65,32,60,73,02,69
310 DATA 67,67,73,79,70,32,32
320 DATA 255,55,17,16,39,33,62
330 DATA 47,60,237,02,40,251,205
340 DATA 229,160,17,232,3,60,237
350 DATA 02,40,231,205,229,160,17
360 DATA 100,040,237,02,40,251
370 DATA 205,229,160,17,10,0,60
380 DATA 237,02,40,251,205,229,160
390 DATA 133,205,229,160,201,205,90
400 DATA 107,62,47,32,1,35,25
410 DATA 35,201,0,0,0,0,0

```



Por que tus dedos no recorren el trabajo duro, M. H. AUS
TRAD la hace por ti. Todos los listados que incluyen este logotipo se encuentran a tu disposición en un cassette mensual, solicitados.

LA MEJOR
ENCICLOPEDIA
INFORMATICA
DEL MUNDO

EL UNICO
SOFTWARE
CONCEBIDO
PARA OPERAR
SIMULTANEAMENTE
EN

5 idiomas:

CASTELLANO
CATALAN
EUSKERA
FRANCES
INGLES

Dilogic, s.a
SOFTWARE EDUCATIVO

TITULOS

Sistema Circulatorio - El Corazón. Organos Reproductores. Sistema Reproductor (Ovulación - Menstruación - Fecundación). Sentidos. Sistema Respiratorio. Las Células. La Sangre. Aparato Digestivo. Hígado - Vesícula Biliar y Páncreas. Aparato Urinario. Sistema Oseo - Huesos. Sistema Muscular. Sistema Nervioso. Sistema Endocrino 1.. Sistema Endocrino 2.. El Cerebro. El Crecimiento. El Embarazo. Las Infecciones. La Energía del Organismo.



AMSTRAD CPC 6128

Londres, 54 08036 Barcelona Tel. 230 94 47 - 230 93 21

Lo prometido es deuda. Aquí está la «nueva entrega» de vuestras carátulas, con otros nombres en portada para que podáis darles diferentes usos.



EXPO-ELECTRONICA'86

Lo que hay que ver.

ERBE
Software



La electrónica:
un mundo en continuo avance.
TV y video, HI-FI, microinformática...
Descubra las últimas novedades
en Expo-Electrónica'86.

- HASTA 24 MESES Y SIN ENTRADA.
- SORTEO DE UN EQUIPAMIENTO COMPLETO DE IMAGEN, SONIDO Y MICROINFORMATICA.

El Corte Inglés

Hay cosas destinadas al éxito: una película de SPIELBERG, un disco de POLICE ó...un juego de ERBE SOFTWARE.

Ordena tus propias ideas

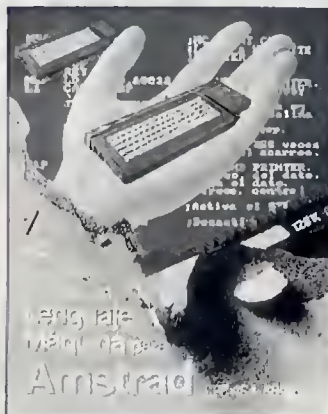
Le sacarás partido a tu ordenador



PROGRAMACION AVANZADA DEL AMSTRAD

Don Thomasson

1.166 ptas.



RUTINAS EN LENGUAJE MAQUINA PARA AMSTRAD

Joe Pritchard

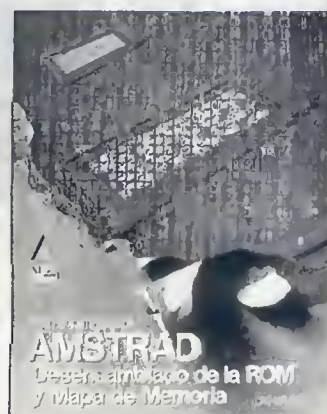
1.590 ptas.



EL LIBRO GIGANTE DE LOS JUEGOS PARA AMSTRAD

K. Bergin y A. Lacey

1.378 ptas.



AMSTRAD: DESENSAMBLADO DE LA ROM Y MAPA DE MEMORIA

Don Thomasson

1.590 ptas.

Otros títulos

PROGRAMACION DEL Z80

Rodnay Zaks

2.915 ptas.

SISTEMAS EXPERTOS

Introducción al diseño y aplicaciones

Tim Hartnell

2.120 ptas.

SIMULACIONES

Replica la realidad con tu ordenador

Tim Hartnell

1.643 ptas.

INTELIGENCIA ARTIFICIAL

Conceptos y programas

Tim Hartnell

1.484 ptas.

EL SUPERLIBRO DE LOS JUEGOS PARA ORDENADOR

Tim Hartnell

2.120 ptas.

CODIGOS Y CLAVES SECRETAS

Criptografía en Basic

Gareth Greenwood

1.378 ptas.

☐ Les ruego me envíen el catálogo de su editorial.

☐ Les ruego me envíen los siguientes títulos:

TOTAL _____

☐ Adjunto talón bancario a
GRUPO DISTRIBUIDOR EDITORIAL, S. A.

☐ Pagaré contrarrembolso (+ 140 pesetas de gasto de envío).

Nombre _____

Profesión _____

Dirección _____

C. P. _____ Localidad _____

Provincia _____

MA

ANAYA
MULTIMEDIA

Adquiéralos en su librería habitual.

Si no le es posible o desea que le enviemos nuestro catálogo, envíe este cupón a:

Apdo. de Correos 14632, Ref. D. de C. 28080 MADRID

Comercializa: GRUPO DISTRIBUIDOR EDITORIAL.

DESARROLLO DE PROGRAMAS

Por José María Díaz

El desarrollo y diseño de aplicaciones es una tarea sumamente importante y delicada. Cualquier herramienta que nos la facilite será, sin duda, bienvenida y merecedora de estudio.



Imaginemos que estamos en plena explosión creadora, frente al teclado del ordenador, intentando crear algo bello y de utilidad: un programa, NUESTRO PROGRAMA.

Ya, desde luego, me he visto en esa situación, tan difícil de explicar con palabras, y confieso que se trata de algo duro.

Concretamente, siempre he echado de menos «algo» que me permitiera, entre otras cosas, echar un vistazo a varias partes del listado simultáneamente.

Los programadores seguro que me entenderán; cuando una aplicación comienza a crecer más allá de nuestras primitivas previsiones, cosa nada rara, y el listado ocupa varias pantallas, es complicado y tedioso localizar un error que se detecta en una parte del programa, pero cuya causa está en otro lugar (las subrutinas, ¿recuerdan?).

Resulta que cada **Amstrad**, de fábrica, incluye una solución para este problema: las ventanas.

Sigamos imaginando, ahora, que a alguien se le ocurre usarlas para dividir la pantalla en, digamos, cuatro partes, de tal forma que en cada una de ellas se pueda ver el listado del programa, totalmente o desde la línea que queramos. Problema solucionado.

Si además el programa crea más herramientas en forma de teclas de función, y se construye como una subrutina, para «mergear-la» con lo que estemos creando, pues mejor que mejor.

Toda esta es lo que nuestro colaborador Robin Nixon ha hecho, en un programa sorprendentemente pequeño.

Robin, tienes la palabra.

Uno de los grandes problemas que uno se encuentra cuando desarrolla software sin la ayuda de una impresora, es la imposibilidad de comparar una parte del programa con otra, puesto que sólo es posible ver un número limitado de líneas en la pantalla a la vez.

La programación sería más fácil si una pu-

diera ver una subrutina y, a la vez, desde dónde se la llama o a dónde se dirige.

Aquí es donde Windows (Robin, sorprendentemente, ha dado un nombre inglés a su programa; aquí diríamos Ventanas con toda modestia) demuestra su utilidad. Funciona definiendo un número de ventanas de texto, que pueden volverse activas pulsando una sola tecla.

Ventanas te da 10 funciones, una en cada tecla de función. Las primeras 4 permiten dirigir la salida de texto a cualquiera de las ventanas presentes en pantalla. Las 4 siguientes son:

- CLS: limpia la ventana activa
- LIST: lista el programa desde el comienzo en la ventana activa.
- LIST N: lo mismo que el anterior, pero a partir de un determinado número de línea (sí, es N).
- RUN: arranca el programa.

Las dos funciones que restan te permiten programar usando el «entorno» Windows a de manera normal.

La forma más eficiente de usar este programa es teclearlo y salvarlo en cinta/disco (!). Cuando lo necesites, simplemente teclea.

MERGE «VENTANAS»

asegurándote, por supuesto, de que tu programa no pasee líneas con el mismo número que esta subrutina.

Para ejecutar el programa, teclea:

GOTO 65000

Con estas palabras, Robin y yo nos despedimos de todos los lectores, en el convencimiento de que queda planteada, en estas páginas, una utilidad verdaderamente útil, ¿para variar?

NOTA: en caso de problemas con la instrucción MERGE, sálvese el programa la primera vez en formato ASCII, tecleando:

SAVE «VENTANAS», A

ESTRUCTURA DEL PROGRAMA

6500-65090	Inicializa las teclas de función 0-9.
65100	Activa la ventana 0 y dibuja la pantalla.
65110	Datos para la información de la parte superior de la pantalla.
65120-65150	Creación de las cuatro ventanas.

```

10 REM VENTANAS
20 REM AMSTRAD SEMANAL
30 REM POR ROBIN NIXON
40 REM
65000 KEY 129, "GOTO 65120"+CHR$(13)
65010 KEY 129, "GOTO 65130"+CHR$(13)
65020 KEY 130, "GOTO 65140"+CHR$(13)
65030 KEY 131, "GOTO 65150"+CHR$(13)
65040 KEY 132, "CLS"+CHR$(13)
65050 KEY 133, "LIST"+CHR$(13)
65060 KEY 134, "LIST "+CHR$(13)
65070 KEY 135, "RUN"+CHR$(13)
65080 KEY 136, "CALL &BC02:MODE 1:CL
65090 KEY 137, "GOTO 65000"+CHR$(13)
S"+CHR$(13)

```

```

65090 KEY 137, "GOTO 65000"+CHR$(13)
65100 MODE 2:ORDER 0:INK 0,0:INK 1
20:CLG 1:RESTORE 65110:FOR Z%=1 TO
4:READ Z%,Z%,Z%,Z%,Z%:WINDOW Z%,
Z%,Z%,Z%:CLS:NEXT:FOR Z%=2 TO 74
STEP 8:WINDOW Z%,Z%+5,2,3:PRINT "
;(Z%-2)/8:READ Z1$:PRINT Z1$:NEXT:
GOTO 65120
65110 DATA 2,41,5,19,43,79,5,11,43,
79,13,19,2,79,21,24, " W0", " W1",
" W2", " W3", " CLS", " LIST", "LIST N
", " RUN", " NORM", " INIT"
65120 WINDOW 2,41,5,19:END
65130 WINDOW 43,79,5,11:END
65140 WINDOW 43,79,13,19:END
65150 WINDOW 2,79,21,24:END

```


EDICION DE RECIBOS (I)

Justo Plá Claver

El programa que presenta hoy para PCW, está realizado en Basic y para un ordenador PCW. No es en realidad un programa, sino una aplicación, es decir, una serie de programas encaminados a un solo fin. Realizar la emisión de recibos, en base a una cartera de clientes.



Para el correcto funcionamiento del mismo, es necesario realizar un fichero para redefinir algunas teclas, que figura en la lista de programas con el nombre KEYS.WP, y ejecutarlo antes de correr el programa con la instrucción SETKEYS KEYS.WP. Además, es necesario cargar los programas en el disco virtual M, para poder tener en la disketera el disco de archivos, y así optimizar su utilización.

Una vez cargado, el programa presenta un menú, que es el que carga los diferentes programas de la aplicación. Las opciones de las que se compone el programa son las habituales de crear los ficheros, borrarlos, introducir los datos de los clientes, imprimir los recibos y realizar un listado por bancos.

El manejo del programa de mantenimiento de clientes, es un poco enrevesado, por lo que es necesario explicarlo con cuidado:

Consultar un cliente: introducir el número de código y pulsar la tecla CAN. Borrado de un cliente: introducir el número de código y pulsar la tecla CORT. Alta o modificación: introducir el número de código y pulsar la tecla COPIA. El cursor se sitúa en la línea siguiente. Si se va a modificar o introducir un dato en esa línea, pulsar INTRO. Si se está dando de alta, es necesario realizar este paso en TODAS las líneas de datos. Si se está modificando, sólo en la de los datos a modificar. Una vez introducido el dato en la línea correspondiente, pulsar INTRO o RETURN, y mover el cursor con las teclas de cursor a la siguiente línea de datos. Una vez introducidos todos los datos, pulsar la tecla SAL, y grabarlo con la tecla INS. La edición de recibos, permite introducir la fecha de expedición, la de vencimiento, la localidad de expedición, el concepto, el nombre del expedidor, y el intervalo de clientes que se desea al emitir los recibos.

Además, el programa permite realizar listado de los recibos emitidos por cada banco, y un listado del impuesto sobre el valor añadido.

En resumen, una aplicación realizada con cuidado, que es perfectamente útil para aquellos que tengan que realizar recibos o menudos, y además con la ventaja de poder modificar la impresión a voluntad al llegar el momento de teclearlo.

PROGRAMA MEN UB

```

10 '----- PROGRAMA MENU.B
AS
20 GOSUB 600
30 GOSUB 480
40 REM --- Impresion de pantalla ---
50 PRINT FNLOC$(10,4);"M E N U  G B
  E R A L"
60 PRINT FNLOC$(20,8);"f1 BORRADO
  DE ARCHIVOS"
70 PRINT FNLOC$(20,10);"f2 - CREACI
  ON DE ARCHIVOS"
80 PRINT FNLOC$(20,12);"f3 - MANTEN
  IMIENTO DE CLIENTES"
90 PRINT FNLOC$(20,14);"f4 - EXPEDI
  CION DE RECIBOS"
100 PRINT FNLOC$(20,16);"f5 - LISTA
  DOS DE BANCOS"
110 PRINT FNLOC$(20,18);"f6 - FIN D
  E TAREA"
120 PRINT FNLOC$(5,24);"Una vez car
  gado el menu quita el disco de prog
  rama del impulsor A e inserta"
130 PRINT FNLOC$(5,25);"al disco de
  datos."
140 PRINT FNLOC$(5,26);"PULSA LA TE
  CLA DE FUNCION DESBADA : :func=IN
  PUTS(1)
150 IF func="1" OR func="6" THEN 14
  0
160 func=VAL(func)
170 ON func GOSUB 340,350,360,370,38
  0,390
180 GOSUB 700
190 PRINT FNLOC$(5,28);"CORRECTO (1
  ES) CANCELAR (CAN) : "
200 GOSUB 600
210 IF blan%1 THEN 250
220 pri.lin%=8:ult.lin%=18:pri.col%
  =16:ult.col%=19
230 GOSUB 400
240 GOTO 140
250 ON func GOTO 270,290,300,310,320
  ,330
260 GOTO 30
270 ERA a.c.e.*
280 GOTO 30
290 CHAIN "crea.bas"
300 CHAIN "manten.bas"
310 CHAIN "edita.bas"
320 CHAIN "listado.bas"
330 SYSTEM
340 PRINT FNLOC$(16,8);"--":RETURN
350 PRINT FNLOC$(16,10);"--":RETURN
360 PRINT FNLOC$(16,12);"--":RETURN
370 PRINT FNLOC$(16,14);"--":RETURN
380 PRINT FNLOC$(16,16);"--":RETURN
390 PRINT FNLOC$(16,18);"--":RETURN
400 REM ----- borrado d
  e la pantalla -----
410 REM pri.lin%=1 linea ult.lin%
  =ultima linea pri.col%=1 columna
  ult.col
  %=ultima columna
420 REM
430 REM
440 FOR pl=pri.lin% TO ult.lin%
450 PRINT FNLOC$(pri.col%,pl%);SPC(
  ult.col%-pri.col%)
460 NEXT pl%
470 RETURN
480 REM ----- marco de la p
  antalla -----
490 lls=CHR$(134)+STRING$(87,CHR$(1
  38))+CHR$(140)
500 PRINT c1s;lls
510 FOR pl=1 TO 28
520 PRINT CHR$(133);STRING$(87,CHR$(
  132))+CHR$(133)
530 NEXT pl
540 PRINT CHR$(135);STRING$(87,CHR$(
  138))+CHR$(141)
550 FOR pl=1 TO 1
560 PRINT CHR$(133);STRING$(87," ")
  ;CHR$(133)
570 NEXT pl
580 PRINT CHR$(131);STRING$(87,CHR$(
  138))+CHR$(137)
590 RETURN
600 REM ----- correct
  o a/n -----
610 rs="":WHILE rs<>"S" AND rs<>"E"
620 rs=INPUT$(1)
630 Wend
640 IF rs="S" THEN blan%1 ELSE blan%
  0
650 RETURN
660 REM ----- inicializacion -----
670 c1s=CHR$(27)+"E"+CHR$(27)+"Y"
680 DEF FN LOC$(x,y)=CHR$(27)+"Y"+C
  HR$(32+y)+CHR$(32+x)
690 RETURN
700 REM ----- limpia linea mensaje
710 PRINT FNLOC$(5,28);"
  " :RETURN
  
```

MANUAL PARA LA APLICACION "EDICION DE RECIBOS"

—Modificación/alta. Intraducir número de código y pulsar (COPIA), entonces el cursor se pondrá en el primer campo de datos. Para moverlo utilizar los cursores arriba/abajo y pulsar (INTRO) o (RETURN) para conectar la línea. Escribir y pulsar (INTRO) o (RETURN) para desconectar la línea. Para finalizar la modificación pulsar (SAL). «El alta de un cliente se considera una modificación de un campo vacío. Sólo se ha de tener en cuenta llenar todos los campos aunque sólo sea conectar y desconectar la línea.»

—Grabar. Una vez modificado un registro ha de grabarlo con la tecla (INS).

—Listado de bancos. Dentro de esta opción se encontrará con una subopción que le permitirá escoger entre un listado para bancos a un listado para Hacienda. En el listado para bancos rellene los campos que le pedirá el ordenador. En el campo de «primer número de cliente:» ha de introducir un código que exista obligatoriamente. Si pulsa (INTRO) o (RETURN) en cualquier campo sin entrar ningún carácter automáticamente subimos al campo anterior. En el listado no aparecerán los clientes que en los cuales el campo de Domiciliación contenga la palabra «contado».

En el listado de IVAS el funcionamiento es igual que el anterior. Las fechas que intraduzcamos deben ir en el formato (DDMMAA) sin espacios ni caracteres entre los dígitos.

—Fin de tarea. Nos devuelve el sistema operativo.

Para valver al menú desde cualquier programa se ha de pulsar (INTRO) o (RETURN) en el primer campo de datos.

NOTA

Teclear los programas independientemente y grabarlos con el nombre que acompaña a cada listado en la sentencia REM de la primera línea del programa.

La semana próxima daremos las siguientes listados junto a unas instrucciones absolutamente necesarias para su correcta utilización.



Programa PCW

PROGRAMA LIST

```

10 '----- PROGRAMA : LIST
ADD.BAS -----
20 ON ERROR GOTO 2300
30 GOSUB 2240
40 GOSUB 2060
50 PRINT FNLOC$(5,28);"Listado (B)a
  nuario - Listado (I)va : :rs=INPU
  TS(1)
60 IF rs="B" OR rs="b" THEN 90
70 IF rs="I" OR rs="i" THEN CHAIN "
  listadoI
80 GOTO 50
90 total=0:contador=0
100 PRINT FNLOC$(10,4);"L I S T A D
  O  B A N C A R I O "
110 PRINT FNLOC$(50,8);"Fecha de ex
  pedicion : "
120 PRINT FNLOC$(50,10);"Fecha de v
  encimiento : "
130 PRINT FNLOC$(20,14);"Entidad Ba
  ncaria : "
140 PRINT FNLOC$(20,20);"Numero pri
  mer cliente : "
150 PRINT FNLOC$(20,22);"Numero ult
  imo cliente : "
160 x%=73:y%=B:alfa%=2:cap%=6:dec%=
  0:GOSUB 780
170 IF ts="" THEN CHAIN "menu
180 fecha%=ts:GOSUB 1090
190 IF fecha%="fallo!" THEN 160 ELS
  E fecha%=fecha%
200 PRINT FNLOC$(73,5);fecha%:" "
210 x%=73:y%=10:alfa%=2:cap%=6:dec%
  =0:GOSUB 780
220 fecha%=ts:GOSUB 1090
230 IF ts="" THEN 160
240 IF fecha%="fallo!" THEN 210 ELS
  E fecha%=fecha%
250 PRINT FNLOC$(73,10);fecha%:" "
260 x%=46:y%=14:alfa%=3:cap%=40:dec%
  =0:GOSUB 780
270 IF ts=STRING$(30," ") THEN 210
  ELSE concept%=ts
280 PRINT FNLOC$(46,14);concept%:"
  "
290 x%=46:y%=20:cap%=3:dec%=0:alfa%
  =2:GOSUB 780
300 IF ts="" THEN 260 ELSE cod1s=ST
  RING$(3-LEN(ts),"0")+ts
310 PRINT FNLOC$(46,20);cod1s:" "
320 x%=46:y%=22:alfa%=2:cap%=3:dec%
  =0:GOSUB 780
330 IF ts="" THEN 290 ELSE cod2s=ST
  RING$(3-LEN(ts),"0")+ts
340 PRINT FNLOC$(46,22);cod2s:" "
350 PRINT FNLOC$(5,28);"CORRECTO (1
  NS) CANCELAR (CAN) : " :GOSUB 2180
360 PRINT FNLOC$(10,2);" "
370 GOSUB 2280
380 IF blan%=0 THEN 320
390 cad$=STRING$(128,CHR$(0)):nomf1
  dat$="a:cclien.tex":nomf1nd$="a:cc
  lien.key":numf1%=1
400 ran%=1:cla%=cod1s:tip%=1
410 GOSUB 1230
420 IF ferror%<0 THEN 2300
430 nombrs=MID$(cad$,1,30)
440 direcciones=MID$(cad$,31,30)
450 poblacions=MID$(cad$,61,20)
460 domiciliacions=MID$(cad$,81,40)
470 importe=CVS(MID$(cad$,121,4))
480 cod1s=MID$(cad$,125,3)
490 GOSUB 690:IF INSTR(domiciliacio
  ns,"Contado")=0 THEN GOSUB 850
500 cad$=STRING$(128,CHR$(0)):nomf1
  dat$="a:cclien.tex":nomf1nd$="a:cc
  lien.key":numf1%=1
510 ran%=1:cla%=cod1s:tip%=2
520 GOSUB 1230
530 IF ferror%<0 AND ferror%<103
  THEN 2300
540 IF ferror%=103 THEN GOSUB 740:G
  OTO 30
550 nombrs=MID$(cad$,1,30)
560 direcciones=MID$(cad$,31,30)
570 poblacions=MID$(cad$,61,20)
580 domiciliacions=MID$(cad$,81,40)
590 importe=CVS(MID$(cad$,121,4))
600 cod1s=MID$(cad$,125,3)
610 IF cod1s=cdn2s THEN GOSUB 740:G
  OTO 30
620 contador=contador+1: IF contado
  r=60 THEN LPRINT CHR$(12):contador=
  0:GOSUB 690
630 IF INSTR(domiciliacions,"Contad
  o")=0 THEN GOSUB 850 ELSE contador=
  
```



```

contador=1
640 GOTO 500
650 REM ----- impresion
660 LPRINT USING "R";R;codis;noebres;:LP
RINT USING "R"
R"doecilicacions;:LP
RINT USING" RR,RRR";INT(1aporte+12/
100)-importe
670 total=total+importe+INT(1aporte
+12/100)
680 RETURN
690 LPRINT " " :conceptos:LPRIN
T "
:fechas;: " * Fecha de creacion : "
:fechas;: " * Fecha de vencimiento :
":fechas;: " *
700 LPRINT
710 LPRINT "COD Nomb
ra Domici
liacion Import"
720 LPRINT "-----"
"
730 RETURN
740 REM ----- linea totalee
750 LPRINT "-----"
760 LPRINT USING "
RRR,RRR";total:L
PRINT CHR$(12)
770 RETURN
780 REM ----- control de digito
s-----
790 REM x=columna y=file
cap=capacidad
800 REM dec=decimales alfa
=numero(1) alfanumericos (2) alfa
numcom (3)
810 REM -----
820 PRINT FNLOC(x,y):STRING$(cap
%dec%+1," ")
830 ts="":t=0:rs="":fallo=0:punto%
=0
840 PRINT CHR$(7)
850 xl=x%
860 PRINT FNLOC(xl,y%):rs=1:PUTS
(1)
870 r=ASC(rs)
880 IF r=13 THEN 950
890 IF r<32 OR r>255 THEN 860
900 IF r=127 AND t=0 THEN t=t-1:ts=
LEFT$(ts,t):PRINT FNLOC(x,y):ts
,"":GOTO 860
910 IF r=127 AND t=0 THEN 860
920 t=t+1
930 IF t>cap%+dec% THEN t=t-1:GOTO
860
940 ts=ts+rs:PRINT FNLOC(x,y):ts
:GOTO 860
950 REM ----- validacion
-----
960 IF alfa=2 THEN RETURN
970 IF alfa=3 THEN ts=ts+STRING$(c
ap%-LEN(ts),""):RETURN
980 FOR numerico=1 TO LEN(ts)
990 rs=MID$(ts,numerico,1)
1000 IF (rs<"0" OR rs>"9") AND rs<>
"." THEN fallo=1
1010 IF rs="." THEN punto=punto%+1
1020 NEXT numerico%
1030 IF fallo=1 OR punto%>1 THEN 7
80
1040 tr=CDBL(VAL(ts))
1050 tr=tr*10^dec%
1060 tr=INT(tr)
1070 tr=tr/10^dec%
1080 RETURN
1090 REM ----- fecha
-----
1100 REM la fecha siempre 6
digitos
1110 REM -----
1120 IF LEN(fechas)>6 THEN fechas=
"falla!":RETURN
1130 f1=VAL(MID$(fechas,1,2))
1140 f2=VAL(MID$(fechas,3,2))
1150 f3=VAL(MID$(fechas,5,2))
1160 IF f1<1 OR f2<1 OR f3>31 OR f2
>12 THEN fechas="falla!":RETURN
1170 f4=(f2=1)+(f2=3)+(f2=5)+(f2=7)
+(f2=8)+(f2=10)+(f2=12)
1180 IF f1>30 AND NOT f4 THEN fecha
s="falla!":RETURN
1190 IF f1>29 AND f2=2 THEN fechas=
"falla!":RETURN
1200 IF f1>28 AND f2=2 AND f3 MOD 4
=0 THEN fechas="falla!":RETURN
1210 fechas=MID$(fechas,1,2)+MID
$(fechas,3,2)+MID$(fechas,5,2)
1220 RETURN
1230 REM ----- f
echeros
-----
1240 REM
1250 REM
1260 REM
1270 REM
1280 REM
1290 REM
1300 REM -----
1310 REM
1320 REM inicio
1330 BUFFERS 10
1340 longreg%=LEN(cads)
1350 OPEN "K",numf1,nomfidat$,nomf
ind$,2,longreg%
1360 longreg%=LEN(cads)
1370 FIELD numf1,longreg% AS fics
1380 ON t:tip% GOSUB 1430,1510,1620,1
730,1800,1890
1390 IF tip%<1 OR tip%>8 THEN 1420
1400 res%=CONSOLIDATE(numf1%)
1410 CLOSE numf1
1420 RETURN
1430 REM lectura de una clave
1440 res%=SEEKKEY(numf1%,0,ran%,cla
s)
1450 IF res%<>0 THEN ferror%=res%:R
ETURN
1460 ferror%=0
1470 num%=FETCHREC(numf1%)
1480 GET numf1%,num%
1490 cads=fics

```

```

1500 RETURN
1510 REM --- lectura de la clave si
guiente
1520 res%=SEEKKEY(numf1%,0,ran%,cla
s)
1530 IF res%<>0 THEN ferror%=res%:R
ETURN
1540 ferror%=0
1550 res%=SEEKNEXT(numf1%,0)
1560 IF res%<>0 AND res%<101 AND r
es%<102 THEN ferror%=res%:RETURN
1570 ferror%=0
1580 num%=FETCHREC(numf1%)
1590 GET numf1%,num%
1600 cads=fics
1610 RETURN
1620 REM lectura de la clave anteri
or
1630 res%=SEEKKEY(numf1%,0,ran%,cla
s)
1640 IF res%<>0 THEN ferror%=res%:R
ETURN
1650 ferror%=0
1660 res%=SEEKPREV(numf1%,0)
1670 IF res%<>0 AND res%<101 AND r
es%<102 THEN ferror%=res%:RETURN
1680 ferror%=0
1690 num%=FETCHREC(numf1%)
1700 GET numf1%,num%
1710 cads=fics
1720 RETURN
1730 REM ----- borrar un registro
1740 res%=SEEKKEY(numf1%,0,ran%,cla
s)
1750 IF res%<>0 THEN ferror%=res%:R
ETURN
1760 ferror%=0
1770 res%=DELKEY(numf1%,0)
1780 IF res%<>0 AND res%<101 AND r
es%<102 AND res%<103 THEN ferror%
=res%:RETURN
1790 ferror%=0:RETURN
1800 REM ----- altas de claves
1810 res%=SEEKKEY(numf1%,0,ran%,cla
s)
1820 IF res%=0 THEN ferror%=900:REI
URN
1830 ferror%=0
1840 LSET fics=cads
1850 res%=ADDREC(numf1%,0,ran%,cla
s)
1860 IF res%<>0 THEN ferror%=res%:R
ETURN
1870 ferror%=0
1880 RETURN
1890 REM ----- modificacion de reg
istros
1900 res%=SEEKKEY(numf1%,0,ran%,cla
s)
1910 IF res%<>0 THEN ferror%=res%:R
ETURN
1920 ferror%=0
1930 num%=FETCHREC(numf1%)
1940 GET numf1%,num%
1950 LSET fics=cads
1960 PUT numf1%
1970 RETURN
1980 REM ----- borrado
de la pantalla -----
1990 REM pri.lin%-1 linea ult.lin
%-ultima linea pri.col%-1 columna
2000 REM ult.co
l%-ultima columna
2100 REM -----
2020 FOR pl=pri.lin% TO ult.lin%
2030 PRINT FNLOC$(pri.col%,pl%):SPC
(ult.col%-pri.col%)
2040 NEXT pl%
2050 RETURN
2060 REM ----- marco de la
pantalla -----
2070 l1=CHR$(134)+STRING$(87,CHR$(
138))+CHR$(140)
2080 PRINT c1;l1;l1
2090 FOR pl=1 TO 28
2100 PRINT CHR$(133);STRING$(87,CHR
$(32));CHR$(133)
2110 NEXT pl
2120 PRINT CHR$(135);STRING$(87,CHR
$(138));CHR$(141)
2130 FOR pl=1 TO 1
2140 PRINT CHR$(133);STRING$(87," "
);CHR$(133)
2150 NEXT pl
2160 PRINT CHR$(131);STRING$(87,CHR
$(138));CHR$(137)
2170 RETURN
2180 REM ----- correc
to e/n
2190 rs="":WHILE rs<>"S" AND rs<>"N"
2200 rs=INPUT$(1)
2210 WRND
2220 IF rs="S" THEN bien%=1 ELSE bi
en%=0
2230 RETURN
2240 REM ----- inicializacion ---
2250 c1=CHR$(27)+"E"+CHR$(27)+"Y"
2260 DEF FN LOC$(x,y)=CHR$(27)+"Y"+
CHR$(32+y)+CHR$(32+x)
2270 RETURN
2280 REM ----- limpia linea mensaje
s-----
2290 PRINT FNLOC$(5,28);"
":RETURN
2300 PRINT c1
2310 PRINT FNLOC$(10,10);": ERROR
!! : Compruebe si el disco del impu
lsoe 'A' es el correcto."
2320 PRINT FNLOC$(10,12);"Si es cor
recto pulse (INS) en caso contrario
cambielo y pulse (CAN)"
2330 PRINT FNLOC$(10,28)
2340 GOSUB 2180
2350 IF bien%=1 THEN PRINT FNLOC$(1
0,28);"Error !!ERR;" en la linea ":
ERL;". Llame al servicio tecnico.":
FOR g=1 TO 5000:NEXT g:CHAIN "menu
2360 IF ferror/>0 THEN GOTO 40
2370 RESUME 40

```



```

10 ' ----- PROGRAMA : CREA.BA
S-----
20 GOSUB 240
30 GOSUB 120
40 PRINT FNLOC$(28,14);"ESPERE UN M
OMENTO. POR FAVOR..."
50 PRINT FNLOC$(5,28);": No toque
el teclado !!"
60 BUFFERS 10
70 CREATE 1,"a:collen.tex","a:coll
en.key",2,135
80 res%=CONSOLIDATE(1)
90 CLOSE 1
100 CHAIN "menu
110 END
120 REM ----- marco de la p
antalla -----
130 l1=CHR$(134)+STRING$(87,CHR$(1
38))+CHR$(140)
140 PRINT c1;l1;l1
150 FOR pl=1 TO 28
160 PRINT CHR$(133);STRING$(87,CHR$
(32));CHR$(133)
170 NEXT pl
180 PRINT CHR$(135);STRING$(87,CHR$
(138));CHR$(141)
190 FOR pl=1 TO 1
200 PRINT CHR$(133);STRING$(87," ")
;CHR$(133)
210 NEXT pl
220 PRINT CHR$(131);STRING$(87,CHR$
(138));CHR$(137)
230 RETURN
240 REM ----- inicializacion ---
250 c1=CHR$(27)+"E"+CHR$(27)+"Y"
260 DEF FN LOC$(x,y)=CHR$(27)+"Y"+C
HR$(32+y)+CHR$(32+x)
270 RETURN
280 REM ----- limpia linea mensajes
-----
290 PRINT FNLOC$(5,28);"
":RETURN

```


LA INFORMATICA A SU ALCANCE

**PROGRAMAS PARA AMSTRAD Y PC COMPATIBLES
LLEVE EL CONTROL Y GESTION DE SU EMPRESA**

CONTAVISA

AMSTRAD 25.000
MS/DOS 30.000

**CONTABILIDAD
GENERAL**

GESCOVISA

AMSTRAD 25.000
MS/DOS 30.000

**GESTION DE EMPRESA
ALBARANES, FACTURAS,
ALMACEN Y REPRESENTANTES**

STOCVISA

AMSTRAD 15.000
MS/DOS 20.000

**CONTROL DE ALMACEN
EXISTENCIAS, MINIMOS
INVENTARIO PERMANENTE
LISTADOS DE PRECIOS**

VISAJET

AMSTRAD 25.000
MS/DOS 30.000

**GENERADOR DE FICHEROS
PROGRAMAS DE FICHEROS
CUARTA GENERACION
RECIBOS, LISTADOS
ETIQUETAS**

FACILES DE USAR, RAPIDOS Y ECONOMICOS

CENTRO DE DESARROLLO DE SOFTWARE



VALLES INFORMATICA, S.A.

c/ Francesc Layret, 76 Tel. (93) 691 23 11
08290 Cerdanyola del Vallès-Barcelona

SOMOS MAYORISTAS

MICRO-1

EL IVA
LO PAGA MICRO-1

C/ Duque de Sesto, 50. 28009 MADRID
Tel. (91) 275 96 16 - 274 75 02
Metro O'Donnell o Goya (aparcamiento gratuito en Felipe II)

**POR CADA PROGRAMA QUE COMPRES ¡¡GRATIS!!
UNOS CASCOS DE MUSICA ESTEREO
SI TU COMPRA ES SUPERIOR A 800 PTAS.**



SOFTWARE SUPER-REBAJADO
DECATHLON 495 PTAS., BEACH HEAD 495 PTAS.,
SOUTHERN BELLE 495 PTAS., DRAGONTORC 495 PTAS.

COMPATIBLE PC IBM
MONITOR FOSFORO VERDE
DOS UNIDADES DE DISCO
360 K UNIDAD
UNIDAD CENTRAL CON 256 K
TECLADO EN CASTELLANO
169.900 PTAS.
INCLUIDO PAQUETE DE
SOFTWARE DE GESTION

TAPA DE METACRILATO 464	PTAS. 895
CABLE CENTRONICS	3.175
CABLE SEGUNDA UNIDAD D.	1.790
CABLES SEPARADORES 6128	1.975
INTERFACE RS232	9.265
CINTA VIRGEN C15	69
CABLE AUDIO	795
CABLE ADAPTADOR 2 JOYSTICK	2.390
CABLES SEPARADORES 464	1.390
CABLE SEPARADOR 8256	2.900
CABLE RS232	2.500
DISKETTES 3"	735

STREET HAWK	PTAS. 2.300
KNIGHT RIDER	2.300
MIAMI VICE	2.300
MOVIE	2.300
EQUINOX	2.100
«V»	2.100
MILLION II DISCO	3.300
GREEN BERET DISCO	3.000
LAS 3 LUCES G DISCO	3.000
DESERT FOX DISCO	3.000
KNIGHT GAMES	2.300
STAINLESS STEEL	2.300
DESERT FOX	2.600
JAK THE NIPPER	2.300
CAULDRON II	2.300
NEXUS DISCO	3.000
RAMBO+MATCH DAY DISCO	3.300
STEINLESS STEEL DISCO	3.000
KNIGHT RIDER DISCO	3.000
KNIGHT GAMES DISCO	3.000

LAPIZ OPTICO
3.295 PTAS.

CASSETTE ESPECIAL ORDENADOR
3.595 PTAS.

	PTAS.
QUICK SHOT I	1.395
QUICK SHOT II	1.695
QUICK SHOT IX	2.395

IMPRESORAS
20% DE
DESCUENTO
PRECIOS EXCEPCIONALES PARA TU AMSTRAD
CPC-464, CPC-6128, PCW-8256, PCW-8512

RATON DE PANTALLA
CON SOFTWARE 7.900 PTAS.
GRATIS UN SUPLETORIO
TELEFONICO



PEDIDOS CONTRA REEMBOLSO SIN NINGUN GASTO DE ENVIO. TEL. (91) 275 96 16 - 274 75 02 O ESCRIBIENDO A: MICRO-1. C/ DUQUE DE SESTO, 50. 28009 MADRID.

Tiendas y distribuidores grandes descuentos.
Dirigirse a Diproimsa. C/ Galatea, 25. Tel. (91) 274 75 03.

AMSTRAD CPC-6128



- MICROPROCESADOR Z80A.
- 128 K DE MEMORIA RAM (41 K DE USUARIO EN BASIC Y 61 K EN CP/M PLUS)
- 48 K DE MEMORIA ROM QUE INCLUYEN EL LOCOMOTIVE BASIC Y EL SISTEMA OPERATIVO.
- 76 TECLAS, TECLADO NUMERICO Y DE CURSOR INDEPENDIENTE.
- TEXTO EN MONITOR DE 20, 40 U-80 COLUMNAS Y GRAFICOS CON DEFINICION DE HASTA 640 X 200 PUNTOS. 27 COLORES DISPONIBLES.
- HASTA 8 VENTANAS EN PANTALLA.
- GENERACION DE SONIDOS EN 3 VOCES Y 8 OCTAVAS.
- UNIDAD DE DISCO DE 3" (169 K BYTES)
- SISTEMAS OPERATIVOS AMS-DOS Y CPM/PLUS
- CONECTORES PARA IMPRESORA, JOYSTICKS, CASSETTE, SEGUNDA UNIDAD DE DISCO, ETC.

SISTEMA COMPLETO CON MONITOR EN FOSFORO VERDE, MANUAL EN CASTELLANO, GARANTIA OFICIAL AMSTRAD ESPAÑA, DISCO CON SISTEMA OPERATIVO CP/M 2.2 Y LENGUAJE DR. LOGO, DISCO CON SISTEMA OPERATIVO CP/M PLUS (CP/M 3.0) Y UTILIDADES, DISCO CON SIETE PROGRAMAS DE OBSEQUIO

84.900 Pts. + I.V.A.

SISTEMA COMPLETO IGUAL AL ANTERIOR PERO CON MONITOR EN COLOR.

119.900 Pts. + I.V.A.

AMSTRAD[®]
E S P A Ñ A

C/ Aravaca, 22. 28040 Madrid.
Tel. 459 30 01. Télex 47660 INSC E. Fax 459 22 92.
Delegación en Cataluña: C. Tarragona, 110.
Tel. 325 10 58. 08015 Barcelona.

COMO DISEÑAR BASE DE DATOS A MEDIDA EN 4' 28"

Es posible que piense que este es un anuncio de publicidad de un programa, pero vamos a demostrarle que no lo es. Usted podrá realizar con nosotros un fichero personalizado, con todas las opciones que le pediría a un programador para realizarle un programa a medida, como pueden ser las salidas por impresoras de sus datos, la gestión de éstos en las búsquedas, el ordenamiento por el campo que desee, y por supuesto con la gran ventaja que lo ha hecho usted y sabe cómo cambiarlo o añadirle las posibles modificaciones que le surjan en un futuro.



La tecnología avanza a grandes pasos. Se consiguen microprocesadores más rápidos que nos gestionan más y más cantidad de memoria, pero la programación aún tiene sus puntos débiles que no pueden seguir el ritmo de los avances tecnológicos. Tal vez uno de ellos sea los retrasos en la culminación de un programa. El programador, con los datos que usted le da, debe efectuar un análisis del mismo, realizar el diseño y finalmente ejecutarlo y probar que funciona. Con nosotros sólo tiene que pensar lo que le hace falta, anotarlo en un papel y ponerse manos a la obra.

Primeros conocimientos

Antes de comenzar la realización del programa vamos a enunciar las primeras nociones para los usuarios que no son muy duchos en el DBASEII. Para comenzar explicaremos los órdenes que necesitamos para la elaboración del fichero en orden de aparición a medida que surjan.

Debemos tener en cuenta un hecho importante antes de proseguir; para continuar los pasos que a continuación vamos a realizar debe tener el ordenador conectado y cargado el DBASEII. En caso de no tenerlo podrían surgirle problemas a la hora de la realización. Muchas veces le dirían que la mejor forma de aprender es leer y experimentar directamente con el ordenador, hecho que le sugerimos tome al pie de la letra. Si alguna vez notara alguna discrepancia entre los mensajes de la pantalla y lo aquí escrito no se asuste y vuelva unos pasos atrás para ver dónde se pudo cometer el error. Comencemos, pues.

La primera de las órdenes que utilizaremos es «**CREATE**». Es uno de los comandos más importantes en la realización de cualquier base de datos, pues es el encargado de la definición de la base de datos. Cuando utilizamos la orden, debemos acompañarla de un nombre, y si lo omitimos, el propio DBASEII nos pedirá el nombre. Cuando se escriba, si no le colocamos la extensión el Sistema le dota con el nombre de DBF. Ya estamos preparados para poder insertar los campos y definir el fichero, para lo cual nos marca la forma correcta de realizar la entrada de estos campos. A la derecha nos marca el número del campo que estamos utilizando, a continuación debemos escribir el nombre del registro. Este nombre no puede contener como primer dígito un número, ni tampoco ningún carácter que no sean los signos alfabéticos o numéricos (estos últimos sí pueden ir en una posición que no sea la anteriormente dicha), a excepción de los dos


```

DISPLAY STRUCTURE
No database file in use, enter filename: ARCHI
Structure for file: ARCHI.DBF
Number of records: 00001
Date of last update: 00/00/00
Primary use database
Fld Name Type Width Dec
001 NOMBRE C 020
002 FIRMA C 020
003 DISTRIB C 020
004 TIPO C 015
005 CONTROL C 015
006 JUGADORES N 002
007 SGA1 CUY C 015
008 DIFICULTAD N 002
009 ORIGINAL C 015
010 ADICION N 002
011 PUNTOS N 002 002
At Total 11 00129

```

Estructura de la base de datos. Capacidad y longitud exacta que ocupa cada ficha en el disco de datos o fichero de datos.

puntos. Si dejásemos espacios en blanca a bien el nombre fuese demasiado extensa, al concluir la línea el DBASEII nos rechazaría la línea y nos permitiría la posibilidad de repetir la misma.

— La primera letra es la 'C'. Encargada de crear registro alfanuméricos, a sea que admite números y caracteres indistintamente.

— La segunda es la 'N'. Esta letra nos asigna la forma del campo en formato numérico. Sólo acepta números, con los cuales más adelante podamos operar. También a la hora de introducirlos los tabula directamente a la derecha del registro.

— Por última, la 'L'. Es una opción especial, tan sólo trabaja con valores lógicos. Estos valores son los de Verdadera 'Y' o Falso 'N'.

Inmediatamente después indicaremos, separada por una coma, la longitud del registro. Esta será necesaria en toda tipo de registro excepto del lógico, mientras que en el numérico podamos indicar el número de decimales que deseamos introducir. Este último se separará con una coma y le escribiremos el número pertinente del decimal. Para finalizar la creación del fichero, al escribir el último registro, pasaremos al siguiente automáticamente y pulsaremos Return o Intra. El ordenador inmediatamente no preguntará si deseamos introducir datos o no. Si lo deseamos pulsaremos 'Y' y podremos introducir ya la primera ficha. En caso contrario nos devolvería el control al sistema de DBASEII.

Esta orden que a continuación explicaremos, es la segunda y la última necesaria para la realización del programa, aunque luego veamos otras relacionadas con la modificación de éstos en caso de querer ampliarla o simplemente traducir los mensajes. La orden en concreto es «INDEX», y se encarga de crearlos el índice de búsqueda por el registro deseado. Tan sólo veremos la concierne a nuestro programa, para no dar a entender que es cosa complicada de realizar, sino toda la contrario.

El formato que utilizaremos será el siguiente:

INDEX ON <registro> TO <nombre del índice>

El registro se refiere al campo en el cual nos interesa la búsqueda, mientras que el índice es el nombre de un pequeño fichero que crea el propio DBASEII para ir a buscar las fichas cuando se requieran.

El comando siguiente no es necesario para la elaboración del programa. Sin embargo, es práctica conocerlo para posteriores modificaciones, o bien para la corrección de algún error que se pueda producir. Cuando estemos en el sistema (el *puntita a la izquierda*) escribiremos el comando:

MODIFY COMMAND <nombre del fichero a modificar>

Las letras en minúsculas significan que se pueden omitir si se desea, y el nombre del fichero es el programa que tenemos que modificar o ampliar. Esta orden nos dará un listado del programa en cuestión y la posibilidad de salir de él sin modificarlo con Control-Q, y de volverlo a grabar con Control-W. En el caso de abaratar el programa, nos pedirá la conformidad de la anulación del mismo o no.

Creación del programa

Vamos a preparar el disco para trabajar, para lo cual efectuaremos una copia de segu-

```

type type archivo cmd
1 Program : ARCHI.DMD
2 Autor : Francisco G.R.
3 Fecha : 31/01/81
4 Aviso : Copyright 1981. All Rights Reserved By United
5 Notas : Gracias por usar DBASEII
6 Modificado: select, selectnum
7
8 DO WHILE 1
9 SET TALK OFF
10 SET BELL OFF
11 SET COLON OFF
12
13 CLR
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
100
101
102
103
104
105
106
107
108
109
110
111
112
113
114
115
116
117
118
119
120
121
122
123
124
125
126
127
128
129
130
131
132
133
134
135
136
137
138
139
140
141
142
143
144
145
146
147
148
149
150
151
152
153
154
155
156
157
158
159
160
161
162
163
164
165
166
167
168
169
170
171
172
173
174
175
176
177
178
179
180
181
182
183
184
185
186
187
188
189
190
191
192
193
194
195
196
197
198
199
200
201
202
203
204
205
206
207
208
209
210
211
212
213
214
215
216
217
218
219
220
221
222
223
224
225
226
227
228
229
230
231
232
233
234
235
236
237
238
239
240
241
242
243
244
245
246
247
248
249
250
251
252
253
254
255
256
257
258
259
260
261
262
263
264
265
266
267
268
269
270
271
272
273
274
275
276
277
278
279
280
281
282
283
284
285
286
287
288
289
290
291
292
293
294
295
296
297
298
299
300
301
302
303
304
305
306
307
308
309
310
311
312
313
314
315
316
317
318
319
320
321
322
323
324
325
326
327
328
329
330
331
332
333
334
335
336
337
338
339
340
341
342
343
344
345
346
347
348
349
350
351
352
353
354
355
356
357
358
359
360
361
362
363
364
365
366
367
368
369
370
371
372
373
374
375
376
377
378
379
380
381
382
383
384
385
386
387
388
389
390
391
392
393
394
395
396
397
398
399
400
401
402
403
404
405
406
407
408
409
410
411
412
413
414
415
416
417
418
419
420
421
422
423
424
425
426
427
428
429
430
431
432
433
434
435
436
437
438
439
440
441
442
443
444
445
446
447
448
449
450
451
452
453
454
455
456
457
458
459
460
461
462
463
464
465
466
467
468
469
470
471
472
473
474
475
476
477
478
479
480
481
482
483
484
485
486
487
488
489
490
491
492
493
494
495
496
497
498
499
500
501
502
503
504
505
506
507
508
509
510
511
512
513
514
515
516
517
518
519
520
521
522
523
524
525
526
527
528
529
530
531
532
533
534
535
536
537
538
539
540
541
542
543
544
545
546
547
548
549
550
551
552
553
554
555
556
557
558
559
560
561
562
563
564
565
566
567
568
569
570
571
572
573
574
575
576
577
578
579
580
581
582
583
584
585
586
587
588
589
590
591
592
593
594
595
596
597
598
599
600
601
602
603
604
605
606
607
608
609
610
611
612
613
614
615
616
617
618
619
620
621
622
623
624
625
626
627
628
629
630
631
632
633
634
635
636
637
638
639
640
641
642
643
644
645
646
647
648
649
650
651
652
653
654
655
656
657
658
659
660
661
662
663
664
665
666
667
668
669
670
671
672
673
674
675
676
677
678
679
680
681
682
683
684
685
686
687
688
689
690
691
692
693
694
695
696
697
698
699
700
701
702
703
704
705
706
707
708
709
710
711
712
713
714
715
716
717
718
719
720
721
722
723
724
725
726
727
728
729
730
731
732
733
734
735
736
737
738
739
740
741
742
743
744
745
746
747
748
749
750
751
752
753
754
755
756
757
758
759
760
761
762
763
764
765
766
767
768
769
770
771
772
773
774
775
776
777
778
779
780
781
782
783
784
785
786
787
788
789
790
791
792
793
794
795
796
797
798
799
800
801
802
803
804
805
806
807
808
809
810
811
812
813
814
815
816
817
818
819
820
821
822
823
824
825
826
827
828
829
830
831
832
833
834
835
836
837
838
839
840
841
842
843
844
845
846
847
848
849
850
851
852
853
854
855
856
857
858
859
860
861
862
863
864
865
866
867
868
869
870
871
872
873
874
875
876
877
878
879
880
881
882
883
884
885
886
887
888
889
890
891
892
893
894
895
896
897
898
899
900
901
902
903
904
905
906
907
908
909
910
911
912
913
914
915
916
917
918
919
920
921
922
923
924
925
926
927
928
929
930
931
932
933
934
935
936
937
938
939
940
941
942
943
944
945
946
947
948
949
950
951
952
953
954
955
956
957
958
959
960
961
962
963
964
965
966
967
968
969
970
971
972
973
974
975
976
977
978
979
980
981
982
983
984
985
986
987
988
989
990
991
992
993
994
995
996
997
998
999
1000
1001
1002
1003
1004
1005
1006
1007
1008
1009
1010
1011
1012
1013
1014
1015
1016
1017
1018
1019
1020
1021
1022
1023
1024
1025
1026
1027
1028
1029
1030
1031
1032
1033
1034
1035
1036
1037
1038
1039
1040
1041
1042
1043
1044
1045
1046
1047
1048
1049
1050
1051
1052
1053
1054
1055
1056
1057
1058
1059
1060
1061
1062
1063
1064
1065
1066
1067
1068
1069
1070
1071
1072
1073
1074
1075
1076
1077
1078
1079
1080
1081
1082
1083
1084
1085
1086
1087
1088
1089
1090
1091
1092
1093
1094
1095
1096
1097
1098
1099
1100
1101
1102
1103
1104
1105
1106
1107
1108
1109
1110
1111
1112
1113
1114
1115
1116
1117
1118
1119
1120
1121
1122
1123
1124
1125
1126
1127
1128
1129
1130
1131
1132
1133
1134
1135
1136
1137
1138
1139
1140
1141
1142
1143
1144
1145
1146
1147
1148
1149
1150
1151
1152
1153
1154
1155
1156
1157
1158
1159
1160
1161
1162
1163
1164
1165
1166
1167
1168
1169
1170
1171
1172
1173
1174
1175
1176
1177
1178
1179
1180
1181
1182
1183
1184
1185
1186
1187
1188
1189
1190
1191
1192
1193
1194
1195
1196
1197
1198
1199
1200
1201
1202
1203
1204
1205
1206
1207
1208
1209
1210
1211
1212
1213
1214
1215
1216
1217
1218
1219
1220
1221
1222
1223
1224
1225
1226
1227
1228
1229
1230
1231
1232
1233
1234
1235
1236
1237
1238
1239
1240
1241
1242
1243
1244
1245
1246
1247
1248
1249
1250
1251
1252
1253
1254
1255
1256
1257
1258
1259
1260
1261
1262
1263
1264
1265
1266
1267
1268
1269
1270
1271
1272
1273
1274
1275
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
1296
1297
1298
1299
1300
1301
1302
1303
1304
1305
1306
1307
1308
1309
1310
1311
1312
1313
1314
1315
1316
1317
1318
1319
1320
1321
1322
1323
1324
1325
1326
1327
1328
1329
1330
1331
1332
1333
1334
1335
1336
1337
1338
1339
1340
1341
1342
1343
1344
1345
1346
1347
1348
1349
1350
1351
1352
1353
1354
1355
1356
1357
1358
1359
1360
1361
1362
1363
1364
1365
1366
1367
1368
1369
1370
1371
1372
1373
1374
1375
1376
1377
1378
1379
1380
1381
1382
1383
1384
1385
1386
1387
1388
1389
1390
1391
1392
1393
1394
1395
1396
1397
1398
1399
1400
1401
1402
1403
1404
1405
1406
1407
1408
1409
1410
1411
1412
1413
1414
1415
1416
1417
1418
1419
1420
1421
1422
1423
1424
1425
1426
1427
1428
1429
1430
1431
1432
1433
1434
1435
1436
1437
1438
1439
1440
1441
1442
1443
1444
1445
1446
1447
1448
1449
1450
1451
1452
1453
1454
1455
1456
1457
1458
1459
1460
1461
1462
1463
1464
1465
1466
1467
1468
1469
1470
1471
1472
1473
1474
1475
1476
1477
1478
1479
1480
1481
1482
1483
1484
1485
1486
1487
1488
1489
1490
1491
1492
1493
1494
1495
1496
1497
1498
1499
1500
1501
1502
1503
1504
1505
1506
1507
1508
1509
1510
1511
1512
1513
1514
1515
1516
1517
1518
1519
1520
1521
1522
1523
1524
1525
1526
1527
1528
1529
1530
1531
1532
1533
1534
1535
1536
1537
1538
1539
1540
1541
1542
1543
1544
1545
1546
1547
1548
1549
1550
1551
1552
1553
1554
1555
1556
1557
1558
1559
1560
1561
1562
1563
1564
1565
1566
1567
1568
1569
1570
1571
1572
1573
1574
1575
1576
1577
1578
1579
1580
1581
1582
1583
1584
1585
1586
1587
1588
1589
1590
1591
1592
1593
1594
1595
1596
1597
1598
1599
1600
1601
1602
1603
1604
1605
1606
1607
1608
1609
1610
1611
1612
1613
1614
1615
1616
1617
1618
1619
1620
1621
1622
1623
1624
1625
1626
1627
1628
1629
1630
1631
1632
1633
1634
1635
1636
1637
1638
1639
1640
1641
1642
1643
1644
1645
1646
1647
1648
1649
1650
1651
1652
1653
1654
1655
1656
1657
1658
1659
1660
1661
1662
1663
1664
1665
1666
1667
1668
1669
1670
1671
1672
1673
1674
1675
1676
1677
1678
1679
1680
1681
1682
1683
1684
1685
1686
1687
1688
1689
1690
1691
1692
1693
1694
1695
1696
1697
1698
1699
1700
1701
1702
1703
1704
1705
1706
1707
1708
1709
1710
1711
1712
1713
1714
1715
1716
1717
1718
1719
1720
1721
1722
1723
1724
1725
1726
1727
1728
1729
1730
1731
1732
1733
1734
1735
1736
1737
1738
1739
1740
1741
1742
1743
1744
1745
1746
1747
1748
1749
1750
1751
1752
1753
1754
1755
1756
1757
1758
1759
1760
1761
1762
1763
1764
1765
1766
1767
1768
1769
1770
1771
1772
1773
1774
1775
1776
1777
1778
1779
1780
1781
1782
1783
1784
1785
1786
1787
1788
1789
1790
1791
1792
1793
1794
1795
1796
1797
1798
1799
1800
1801
1802
1803
1804
1805
1806
1807
1808
1809
1810
1811
1812
1813
1814
1815
1816
1817
1818
1819
1820
1821
1822
1823
1824
1825
1826
1827
1828
1829
1830
1831
1832
1833
1834
1835
1836
1837
1838
1839
1840
1841
1842
1843
1844
1845
1846
1847
1848
1849
1850
1851
1852
1853
1854
1855
1856
1857
1858
1859
1860
1861
1862
1863
1864
1865
1866
1867
1868
1869
1870
1871
1872
1873
1874
1875
1876
1877
1878
1879
1880
1881
1882
1883
1884
1885
1886
1887
1888
1889
1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902
1903
1904
1905
1906
1907
1908
1909
1910
1911
1912
1913
1914
1915
1916
1917
1918
1919
1920
1921
1922
1923
1924
1925
1926
1927
1928
1929
1930
1931
1932
1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943
1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997
1998
1999
2000
2001
2002
2003
2004
2005
2006
2007
2008
2009
2010
2011
2012
2013
2014
2015
2016
2017
2018
2019
2020
2021
2022
2023
2024
2025
2026
2027
2028
2029
2030
2031
2032
2033
2034
2035
2036
2037
2038
2039
2040
2041
2042
2043
2044
2045
2046
2047
2048
2049
2050
2051
2052
2053
2054
2055
2056
2057
2058
2059
2060
2061
2062
2063
2064
2065
2066
2067
2068
2069
2070
2071
2072
2073
2074
2075
2076
2077
2078
2079
2080
2081
2082
2083
2084
2085
2086
2087
2088
2089
2090
2091
2092
2093
2094
2095
2096
2097
2098
2099
2100
2101
2102
2103
2104
2105
2106
2107
2108
2109
2110
2111
2112
2113
2114
2115
2116
2117
2118
2119
2120
2121
2122
2123
2124
2125
2126
2127
2128
2129
2130
2131
2132
2133
2134
2135
2136
2137
2138
2139
2140
2141
2142
2143
2144
2145
2146
2147
2148
2149
2150
2151
2152
2153
2154
2155
2156
2157
2158
2159
2160
2161
2162
2163
2164
2165
2166
2167
2168
2169
2170
2171
2172
2173
2174
2175
2176
2177
2178
2179
2180
2181
2182
2183
2184
2185
2186
2187
2188
2189
2190
2191
2192
2193
2194
2195
2196
2197
2198
2199
2200
2201
2202
2203
2204
2205
2206
2207
2208
2209
2210
2211
2212
2213
2214
2215
2216
2217
2218
2219
2220
2221
2222
2223
2224
2225
2226
2227
2228
2229
2230
2231
2232
2233
2234
2235
2236
2237
2238
2239
2240
2241
2242
2243
2244
2245
2246
2247
2248
2249
2250
2251
2252
2253
2254
2255
2256
2257
2258
2259
2260
2261
2262
2263
2264
2265
2266
2267
2268
2269
2270
2271
2272
2273
2274
2275
2276
2277
2278
2279
2280
2281
2282
2283
2284
2285
2286
2287
2288
2289
2290
2291
2292
2293
2294
2295
2296
2297
2298
2299
2300
2301
2302
2303
2304
2305
2306
2307
2308
2309
2310
2311
2312
2313
2314
2315
2316
2317
2318
2319
2320
2321
2322
2323
2324
2325
2326
2327
2328
2329
2330
2331
2332
2333
2334
2335
2336
2337
2338
2339
2340
2341
2342
2343
2344
2345
2346
2347
2348
2349
2350
2351
2352
2353
2354
2355
2356
2357
2358
2359
2360
2361
2362
2363
2364
2365
2366
2367
2368
2369
2370
2371
2372
2373
2374
2375
2376
2377
2378
2379
2380
2381
238
```



```

A>type ar-main.cmd

! Program : AR-MAIN.CMD
! Author : Your Name
! Date : 00/00/00
! Notice : Copyright 1900, All Rights Reserved
! Reserved : select, selectnum, addchoice, poschoice,
! error, findkey, expression, string, lastof
! clearline, addchoice, is:doms, lastrecord
!

SET TALK OFF
SET BELL OFF
SET COLON OFF
! ---Use blanks to clear-to-end-of-line
STORE %(STR(0.81),1.60) TO clearline
DO WHILE T

ERASE
@ 1,0 SAY "===== "
@ 1.40 SAY " "
@ 2,0 SAY "AR"

Press RETURN to Continue
ENDIF
USE
CASE selectnum= 4
* DO pack
DO AR-pack
ENDCASE
ENDDO 1
! EOF: AR-MAIN.CMD

006 Jugadores N 002 000
007 Gráficos C 015 000
008 Dificultad N 002 000
009 Original C 015 000
010 Adición N 002 000
011 Puntos N 002 002

Nos preguntará si deseamos introducir fichas,
le responderemos que no para crear el fichero
indexado. Este lo creamos escribiendo:

'INDEX ON Nombre TO ARCHI'

La búsqueda de datos se realizará por el
nombre del programa, pudiéndose modificar
cambiando el 'Nombre' por el campo que deseamos.
Cuando concluya la operación sólo
nos quedo ir mirando la pantalla para contemplar
nuestro programa.

Teclamos ahora la palabra:

/. DO DGEN

A>type ar-frame.cmd

! Program : AR-FRAME.CMD
! Author : Your Name
! Date : 00/00/00
! Notice : Copyright 1900, All Rights Reserved
!

@ 2,0 SAY "===== "
@ 2.40 SAY " "
@ 4,0 SAY "Nombre : "
@ 5,0 SAY "Firma : "
@ 6,0 SAY "Distrib : "
@ 7,0 SAY "Tipo : "
@ 8,0 SAY "Control : "
@ 9,0 SAY "Jugadores : "
@ 10,0 SAY "Gráficos : "
@ 11,0 SAY "Dificultad : "
@ 12,0 SAY "Original : "
@ 13,0 SAY "Adición : "
@ 14,0 SAY "Puntos : "
@ 17,0 SAY "===== "

Press RETURN to Continue
RETURN
! EOF: AR-FRAME.CMD

Máscara de presentación en pantalla de una ficha.

A>

! Program : AR-GETS.CMD
! Author : Your Name
! Date : 00/00/00
! Notice : Copyright 1900, All Rights Reserved
!

IF #
@ 1.55 SAY "DELETED"
ELSE
@ 1.55 SAY " "
ENDIF
@ 4.11 GET Nombre
@ 5.11 GET Firma
@ 6.11 GET Distrib
@ 7.11 GET Tipo
@ 8.11 GET Control
@ 9.11 GET Jugadores
@ 10.11 GET Gráficos
@ 11.11 GET Dificultad
@ 12.11 GET Original
@ 13.11 GET Adición

Press RETURN to Continue
@ 14.11 GET Puntos
RETURN
! EOF: AR-GETS.CMD

Programa encargada de introducir y extraer datos
del fichero por registros.

```

Nos remitirá a un menú y pulsaremos la opción (1). Nos preguntará por el nombre de la base de datos (en inglés nos hará la pregunta) y le escribiremos el nombre de nuestra base de datos 'ARCHI'. Luego nos pedirá el nombre que va a tener este menú, le escribiremos: 'ARCHIVO', y para finalizar le daremos el encabezamiento 'MENU PRINCIPAL'. A continuación nos soldrá:

0. exit
- 1.

Nosotros lo terminaremos de llenar con:

0. exit.
1. Introducir Datos.
2. Listado por Impresora.
3. Listado de Etiquetas.
- 4.

En el último número pulsaremos 'RETURN' para salir. Después de mostrarnos las másco- ra de presentación tal y como quedará, nos realizará tres preguntas: Grabamos, Salimos o Repetimos, si todo está correcto pulsaremos Grabar (SAVE).

Comprobaremos cómo se graba en el disco un programa que es el encargado de realizar la máscara de presentación. Al poco tiempo nos pedirá que pulsemos una tecla y volveremos al menú primero que nos encontramos. Ahora pulsaremos la opción (2).

Con esta opción crearemos la base principal de nuestro programa. Nos preguntará por el nombre de la base de datos, nosotros le daremos 'ARCHI'. Luego nos preguntará por el fichero indexado y le volveremos a dar 'ARCHI'. Para finalizar nos preguntará por el nombre que tiene el campo de búsqueda y le daremos 'NOMBRE'.

En ese momento nos realizará la comprobación si existe algún nombre repetido ya en el directorio, y nos creará todo el programa principal (búsqueda, introducción, etc.). Cuando acabe de presentarnos los programas por la pantalla, nos pedirá que pulsemos una tecla para volver al menú principal.

Una vez en él, pulsaremos la opción (0). Entonces saldremos a sistema DBASEII, en él volveremos a salir y no iremos a CP/M. Una vez en este último borraremos los programas:

MENUGEN.CMD
FILEGEN.CMD

Esto lo hacemos para obtener mayor capacidad en el disco y poder seguir trabajando. Nuevamente cargaremos en memoria el DBASEII y teclaremos 'DO DGEN'. Ahora nos marcharemos a la opción (3).

Con esta opción vamos a crear los listados por impresora de todas las fichas que tengamos en nuestra base de datos. Nos va a realizar más preguntas que las otras opciones, pero no hoy ningún problema si se realiza correctamente. La primera es el nombre de la base de datos 'ARCHI', seguidamente le daremos el nombre del programa que estamos realizando 'LISTADO'. Ahora nos realizará un total de seis preguntas que son:


```

Atype ar-edit.cad
! Program : AR-EDIT.CMD
! Author : Your Name
! Date : 00/00/00
! Notice : Copyright 1990, All Rights Reserved
!
DO WHILE T
  STORE " " TO editchoice
  @ 19.0 SAY "COMMAND: (E)dit, (D)lete, (U)ndele, (P)osition"
  GET editchoice PICTURE "I"
  READ NOUPDATE
  CLEAR GETS
  @ 18.0 SAY clearline
  DO CASE
    CASE editchoice = " "
      RETURN GET select
    CASE editchoice = "P"
      @ ---Position
      STORE "X" TO poschoice
      Press RETURN to Continue
      DO WHILE poschoice = " "
        DO AR-posn
      ENDDO
      CASE editchoice = "D"
        @ ---Delete
        DELETE
        @ 1.75 SAY "DELETED"
      CASE editchoice = "U"
        @ ---Undelete
        RECALL
        @ 1.75 SAY " "
      CASE editchoice = "E"
        @ ---Edit
        @ 10.0 SAY "Press <control-U> to exit"
        IF A < 0
          DO AR-some
          READ
        ENDIF
      CASE editchoice = "C"
        @ ---Continue to the next record
        STORE A TO lastrecord
        CONTINUE
      Press RETURN to Continue
      @ ---Check for END OF FILE
      IF NOT EOF
        DO AR-get
        CLEAR GETS
      ELSE
        @ ---EOF encountered
        GOTO lastrecord
        @ 19.0 SAY clearline
        @ 18.0 SAY "END-OF-FILE encountered"
        STORE " " TO select
        @ 19.0 SAY "Strike any key to continue"
        GET select
      READ NOUPDATE
        @ 18.0 SAY clearline
        @ 19.0 SAY clearline
      ENDDO
    ENDCASE
  ENDDO
  @ (UI: AR EDIT) CMD
  A

```

Gestiona la base de datos.

Margen Izquierdo.
 Líneas por Página.
 Número de Columnas.
 Cabecera de Página.
 Requeridos en todos.
 Requeridos en los subtítulos.

A todas ellas y para no hacer compleja la realización pulsaremos la tecla de RETURN, con lo cual queda el valor por defecto. A continuación nos saldrá un número con dos palabras que significan el tamaño y el contenido, contenido éste que es el nombre que hemos dado a nuestra base de datos por un registro específico, y por supuesto la longitud de éste. En caso de error el ordenador nos repetirá la pregunta para que los volvamos a realizar. Si no nos hemos equivocado nos pedirá el encabezamiento que debe aparecer en el listado de la impresora, que muy bien puede ser distinto o no. Nosotros indicaremos los siguientes campos con sus longitudes:

1. Tamaño, Nombre: 20, NOMBRE
Contenido : NOMBRE
2. Tamaño, Nombre: 20, FIRMA
Contenido : FIRMA
3. Tamaño, Nombre: 15, TIPO
Contenido : TIPO
4. Tamaño, Nombre: 15, GRAFICOS
Contenido : GRAFICOS
5. Tamaño, Nombre: 2, ADICION
Contenido : ADICION
6. Contenido : 2, PUNTOS
Contenido : PUNTOS
7. Tamaño, Nombre:

En el último, y para salir, pulsaremos RETURN. Nuevamente nos marcharemos al menú principal y pulsaremos por último la opción (4). En esta opción vamos a realizar el programa de etiquetas.

De nuevo nos va a realizar las preguntas de nombre de la base de datos 'ARCHI', a continuación el nombre del fichero que vamos a crear 'ETI', y por último escribiremos los nombres que queramos imprimir en formato de etiqueta. Para hacernos el formato de introducción, antes de comenzar hay seis ejemplos en los que vienen reflejadas las opciones posibles. Nuestro formato será el siguiente:

1. NOMBRE
2. [FIRMA] (en caso de no encontrar valor no nos dejaría la línea en blanco, sino que lo uniría con el siguiente valor)
3. TIPO
4. PUNTOS
- 5.

```

Atype ar-pack.cad
! Program : AR-PACK.CMD
! Author : Your Name
! Date : 00/00/00
! Notice : Copyright 1990, All Rights Reserved
!
ERASE
@ 2.0 SAY "P A C K A R C H I"
@ 2.72 SAY DATE()
@ 3.0 SAY " "
@ 3.40 SAY " "
STORE "ND" TO select
@ 5.0 SAY "PACK the entire file? (Y/N) "
GET select PICTURE "I"
READ NOUPDATE
IF select < "YES"
  RETURN
ENDIF
@ 6.0 SAY "ARCHI.OLD will be your backup data file."
IF FILE("ARCHI.OLD")
  STORE " " TO select
  Press RETURN to Continue
  @ 6.10 SAY "Delete the old backup file? (Y/N) "
  GET select PICTURE "I"
  READ NOUPDATE
  IF select < "Y"
    RETURN
  ENDIF
  DELETE FILE ARCHI.DLG
ENDIF
USE
RENAME ARCHI.DLG TO ARCHI.OLD
@ 6.10 SAY " "
USE ARCHI.DLG
SET TALK ON
SET ECHO ON
COPY TO ARCHI
USE
USE ARCHI
@ ---Recreate index file
INDEX ON NONE TO ARCHI
USE
GET ECHO OFF
Press RETURN to Continue
SET TALK OFF
STORE " " TO select
@ 22.0 SAY clearline
@ 22.0 SAY "Strike any key to continue"
GET select
READ NOUPDATE
RETURN
@ (UI: AR-PACK.CMD)

```

Nos ordena el fichero por el compa del nombre, creándonos una vía de seguridad con posibilidad de borrarla, a la hora de su creación.

```

Atype ar-look.cad
! Program : AR-LOOK.CMD
! Author : Your Name
! Date : 00/00/00
! Notice : Copyright 1990, All Rights Reserved
!
STORE " " TO expression.string
@ ---LOCATE option
DO WHILE expression = " "
  @ 10.0 SAY "EXAMPLE: STATE 'CA'"
  @ 18.0 SAY clearline
  @ 17.0 SAY " "
  ACCEPT "Enter LOCATE expression " III expression
  @ 18.0 SAY clearline
  STORE TRIM(expression) TO expression

```

```

DO CASE
  CASE expression = " "
    EXIT
  RETURN
  CASE U = TEST(expression)
  IF AS RETURN to Continue
    @ ---INVALID EXPRESSION
    @ 19.0 SAY clearline
    STORE " " TO select
    @ 19.0 SAY "INVALID EXPRESSION: "
    GET select
    READ NOUPDATE
    @ 19.0 SAY clearline
    STORE " " TO expression
  OTHERWISE
    @ ---LOCATE the record
    @ ---How many file for a faster LOCATE
    SET INDEX TO
    LOCATE FOR expression
    IF NOT EOF
      @ ---Found a matching record
      STORE A TO lastrecord
      @ ---Reopen index file
      SET INDEX TO ARCHI
      GOTO lastrecord
    ELSE
      @ ---No more index file and
      Press RETURN to Continue
      @ ---Reset LOC earlier to true + 1
      SET INDEX TO ARCHI
      NO EDIT
      NO D
      ENDIF
    ENDIF
  IF poschoice = "C"
    @ ---Return to calling program if only the LOCATE was desired
  ENDIF
  @ ---DISPLAY option
  STORE " " TO string
  DO WHILE string = " "
    @ 18.0 SAY "EXAMPLE: NAME+ADDRESS"
    @ 18.0 SAY " "
    ACCEPT "Enter DISPLAY string " III string
    @ 18.0 SAY clearline
    STORE TRIM(string) TO string
    DO CASE
      CASE string = " "
        Press RETURN to Continue
        @ ---Exit
        @ 18.0 SAY clearline
        @ 19.0 SAY clearline
        RETURN
      CASE U = TEST(string)
        @ ---INVALID EXPRESSION
        @ 19.0 SAY clearline
        STORE " " TO select
        @ 19.0 SAY "INVALID DISPLAY EXPRESSION: "
        GET select
        READ NOUPDATE
        @ 19.0 SAY clearline
        STORE " " TO string
      ENDCASE
    ENDIF
  @ ---Now, DISPLAY the expression
  STORE F TO select.isosome
  DO WHILE NOT EOF
    @ ---The following set of default commands are to
    @ ---clear to end-of-screen. If you have an IBM PC,
    @ ---you can replace these commands with the single
    Press RETURN to Continue
    @ ---Command: @ 4.0 ERASE
    @ 4.0 SAY clearline
    @ 5.0 SAY clearline
    @ 6.0 SAY clearline
    @ 7.0 SAY clearline
    @ 8.0 SAY clearline
    @ 9.0 SAY clearline
    @ 10.0 SAY clearline
    @ 11.0 SAY clearline
    @ 12.0 SAY clearline
    @ 13.0 SAY clearline
    @ 14.0 SAY clearline
    @ 15.0 SAY clearline
    @ 16.0 SAY clearline
    @ 17.0 SAY clearline
    @ 18.0 SAY clearline
    @ 19.0 SAY clearline
    @ 20.0 SAY clearline
    @ 21.0 SAY clearline
    @ 22.0 SAY clearline
    @ 20.0 SAY " "
  ENDIF
  Press RETURN to Continue
  @ 20.40 SAY " "
  STORE A TO row
  DO WHILE NOT EOF AND row < 25
    STORE I TO isosome
    @ row.0 SAY string
    STORE row + 1 TO row
    CONTINUE
  ENDDO
  @ ---A logical memory variable is used to detect the
  @ ---end-of-file. The EOF function cannot be used since
  @ ---it is reset by the FIND statement before
  STORE EOF TO i.eof
  IF NOT isosome
    @ ---No matching record
    @ 4.0 SAY "NO MATCHING RECORD"
  ENDIF
  STORE " " TO select
  @ 21.0 SAY "Strike any key to continue"
  GET select
  READ NOUPDATE
  @ 21.0 SAY clearline
  ENDDO
  Press RETURN to Continue
  @ ---The following set of default commands are to
  @ ---clear to end-of-screen. If you have an IBM PC,
  @ ---you can replace these commands with the single
  @ ---command: @ 4.0 ERASE
  @ 4.0 SAY clearline
  @ 5.0 SAY clearline
  @ 6.0 SAY clearline
  @ 7.0 SAY clearline
  @ 8.0 SAY clearline
  @ 9.0 SAY clearline
  @ 10.0 SAY clearline
  @ 11.0 SAY clearline
  @ 12.0 SAY clearline
  @ 13.0 SAY clearline
  @ 14.0 SAY clearline
  @ 15.0 SAY clearline
  @ 16.0 SAY clearline
  @ 17.0 SAY clearline
  @ 18.0 SAY clearline
  @ 19.0 SAY clearline
  @ 20.0 SAY clearline
  Press RETURN to Continue
  @ 21.0 SAY clearline
  @ 20.0 SAY clearline
  NO TALK
  RETURN
  @ (UI: AR-LOOK.CMD)

```



```

er-posn cmd

* Program : AR-POSN CMD
* Author : Your Name
* Date : 00/00/00
* Notice : Copyright 1900. All Rights Reserved
*
STORE " " TO poschoice
@ 18,0 SAY clearline
@ 19,0 SAY "COMMAND: (D)isplay, (F)ind, "+:
"(L)ocate, (C)ontinue, (S)kip ":
GET poschoice PICTURE "I"
READ NOUPDATE
CLEAR GETS
@ 18,0 SAY clearline
IF NOT (poschoice = "OFLCS")
RETURN
ENDIF
IF poschoice = "F"
* ---Find
@ 17,0 SAY "--"
ACCEPT "Enter NUMBER " TO findkey
Press RETURN to Continue
@ 18,0 SAY clearline
STORE TRIM(findkey) TO findkey
IF findkey = " "
RETURN
ENDIF
STORE A TO lastrecord
FIND findkey
IF (A <> 0)
DO AR-gets
CLEAR GETS
ELSE
* ---NO FIND
GOTO lastrecord
@ 18,0 SAY clearline
@ 19,0 SAY " *findkey+ "+" not in index"
STORE " " TO select
@ 19,0 SAY "Strike any key to continue. ":
GET select
READ NOUPDATE
@ 18,0 SAY clearline
@ 19,0 SAY clearline
ENDIF
Press RETURN to Continue
ELSE
* ---S)kip, (C)ontinue, (D)isplay, or (L)ocate
STORE A TO lastrecord
DO CASE
CASE poschoice = "S"
* ---S)kip
SKIP
CASE poschoice = "C"
* ---C)ontinue
CONTINUE
OTHERWISE
* ---D)isplay or (L)ocate
DO AR-locat
IF expression = " "
RETURN
ENDIF
IF poschoice = "D"
IF string = " "
RETURN
ENDIF
DO AR-frame
ENDIF
ENDIF
Press RETURN to Continue
ENDCASE
* ---Check for END-OF-FILE.
IF NOT EOF
DO AR-gets
CLEAR GETS
ELSE
* ---EOF encountered.
GOTO lastrecord
@ 18,0 SAY clearline
@ 19,0 SAY "END-OF-FILE encountered"
STORE " " TO select
@ 19,0 SAY "Strike any key to continue. ":
GET select
READ NOUPDATE
@ 18,0 SAY clearline
@ 19,0 SAY clearline
ENDIF
ENDIF
RETURN
* EOF: AR-POSN CMD

```

Como siempre, para terminar pulsaremos 'RETURN' en la última línea que nos muestre DBASEII. Cuando termine de realizar la grabación en el disco del nuevo fichero, nos dirá de nuevo que pulsemos una tecla para irnos al menú principal. Una vez en este menú pulsaremos la opción (0) para salir, debido a que ya se ha creado el programa con todas sus opciones y distribuciones, vamos a proceder a dejar el disco con la mayor capacidad posible. Cuando estemos en el sistema de DBASEII, saldremos de éste con el comando 'QUIT'.

Una vez en CP/M, borraremos los siguientes programas:

LABELGEN.CMD
FORMGEN.CMD
DGEN.CMD
DGEN.OVL

Terminación y retoque posterior

El programa está listo para ponerlo en la mesa y comérnoslo a base de introducirle datos y más datos. Pero por supuesto hay que darle el toque final para dejarlo 'CHAPÉAU'. Este consiste en introducirnos en el fichero 'AR-MAIN' mediante el comando MODIFY COMMAND AR-MAIN, y una vez dentro de él colocarnos en la página en donde se encuentran los DO CASE. Podremos comprobar que existe un '* DO Introducir Datos', debemos de quitar el asterisco, pues este funcional igual que la sentencia en Basic REMark, o sea que actúa como si no existiera la línea o la orden posterior. Cambiaremos las líneas siguientes:

* DO Introducir Datos.
* DO Listado por Impresoras.
* DO Listado de Etiquetas.

Por las que a continuación exponemos para que se puedan entrelazar los programas, con lo cual quedarán como indicamos a continuación:

DO AR-MAIN
DO LISTADO
DO ETI

Cuando hallamos terminado el proceso de modificación, concluiremos con Control-W, que nos graba el programa ya modificado.

La última observación la tenemos en el programa ARCHIVO, en el cual se encuentra la máscara de presentación, en la primera línea de programa (no los REMARK en el cual nos pide el nombre, la fecha, etc.), en la cual tenemos un SET, hay que intercalar la línea que tenemos en cuarto lugar y se refiere al 'DO WHILE'. La cambiaremos de posición colocándola en primer lugar y eliminándola de

```

A>type ar-some.cmd

* Program : AR-SDME CMD
* Author : Your Name
* Date : 00/00/00
* Notice : Copyright 1900. All Rights Reserved
*
IF *
@ 1,55 SAY "DELETED"
ELSE
@ 1,55 SAY " "
ENDIF
@ 4,11 SAY Nombre
@ 5,11 GET Firma
@ 6,11 GET Distrib
@ 7,11 GET Tipo
@ 8,11 GET Control
@ 9,11 GET Jugadores
@ 10,11 GET Graficos
@ 11,11 GET Dificultad
@ 12,11 GET Original
@ 13,11 GET Adicion

Press RETURN to Continue
@ 14,11 GET Puntos
RETURN
* EOF: AR-SDME CMD
A>type

```

la posición en la que se encuentra, de no hacerlo, cuando pasemos de una opción del menú principal a otra, nos imprimirá en la pantalla unos números que proceden de la basura que queda acumulada en el Buffer.

Las modificaciones las podemos comprobar en el listado que está incluido en este artículo, pudiéndolo transformar y sacar totalmente en castellano todo el programa, tan sólo nos hace falta tener paciencia y valor para realizarlo.

Mensaje de error del DBASEII

Para los que han leído este artículo y desean profundizar un poco en el mundo del DBASEII, vamos a remitirles los errores más significativos que se pueden producir:

BAD FILE NAME: Error en el nombre del Fichero.

BAD NAME FIELD: Nombre no reconocido por el sistema.

BAD TYPE FIELD: El tipo debe ser C, N o L.

DIRECTORY IS FULL: No caben más registros en el disco.

DISCK IS FULL: No caben más programas en el disco.

ILLEGAL VARIABLE NAME: El carácter debe ser alfanumérico y dos puntos en los nombres de variables y campos.

INDEX DOES NOT MATCH DATABASE: No podemos hacer coincidir el nombre calve con el nombre de la base de datos.

OUT OF MEMORY FOR MEMORY VARIABLES: Hemos rebasado el tamaño de las variables o nos hemos excedido en el número de éstas.

RECORD NOT IN INDEX: No hemos actualizado el fichero de INDEX después de introducir un nuevo registro.

RECOR OUT OF RANGE: Hemos introducido un registro superior al número que tenemos en nuestra base de datos.

TOO MANY CHARACTERS: Línea de orden demasiado grande para DBASEII.

TOO MANY FILES ARE OPEN: El límite de ficheros permitidos para abrirse a la vez es de 16, y lo hemos sobrepasado.

TOO MANY MEMORY VARIABLES: Tenemos un máximo de 64 variables en la memoria no pudiéndola superar.

*****UNKNOWN COMMAND:** Comando desconocido para DBASEII.

VARIABLE CANNOT BE FOUND: La variable que está utilizando necesita crearla antes.

Nota: Es posible que al intentar realizar el programa, cuando marca la orden de 'DO DGEN' el ordenador le dé un mensaje de error. Esto es debido a un problema que trajeron las primeras copias que salieron al mercado español, en caso de que suceda, diríjase al distribuidor donde lo adquirió y pídale que se la cambie por estar defectuosa. Damos las gracias por la colaboración desinteresada prestada por MICROWARE para la realización de este artículo.

Francisco G. R.

A:TYPE LISTADD.CMD

```
# Program : LISTADD.CMD
# Author : Your Name
# Date : 00/00/00
# Notice : Copyright 1988, All Rights Reserved
# Reserved : pagenum, line, pagehdg, col:hdg, condition,
# lastrec
```

```
#
SET TALK OFF
SET BELL OFF
SET MARGIN TO 1
STORE 1 TO pagenum
STORE /54 TO line
STORE "LISTADD: FOR IMPRESORA" TO pagehdg
STORE (80-LEN(pagehdg))/2 TO col:hdg
#
# ---Open the datafile and print the report
USE ARCH1
ERASE
# 2. 0 SAY pagehdg
# 2.72 SAY DATE()
```

```
Press RETURN to Continue
# 3. 0 SAY "*****"
# 3.40 SAY "*****"
STORE " " TO select
# 5.0 SAY "Output to the screen or printer? (S/P) "
GET select PICTURE "I"
```

```
READ
DO CASE
CASE select = "S"
ERASE
STORE 22 TO pagelen
CASE select = "P"
SET FORMAT TO PRINT
STORE 56 TO pagelen
OTHERWISE
ERASE
SET BELL ON
SET TALK ON
RETURN
ENDCASE
# ---Enter FOR (expression) for the report, such as,
# ---STORE "STATE = 'CA'" TO condition
STORE " " TO condition
```

```
Press RETURN to Continue
DO WHILE NOT EOF
IF line > pagelen
IF select = "S"
ERASE
ELSE
EJECT
ENDIF
# 0.0 SAY "PAGE NO "
# 0.9 SAY STR(pagenum,3)
# 2.col:hdg SAY pagehdg
#
# -- Generate column headings
# 4. 0 SAY "NUMERO"
# 4. 23 SAY "FIRMA"
# 4. 46 SAY "TIPO"
# 4. 64 SAY "GRAFICOS"
# 4. 82 SAY "AD"
# 4. 97 SAY "PUN"
# 5. 0 SAY "*****"
# 5. 23 SAY "*****"
# 5. 46 SAY "*****"
# 5. 64 SAY "*****"
```

```
Press RETURN to Continue
# 5. 82 SAY "=="
# 5. 87 SAY "=="
STORE pagenum+1 TO pagenum
STORE / TO line
ENDIF
# ---Test to see if the condition exists
IF condition <> " "
IF NOT (&condition)
SKIP
LOOP
ENDIF
ENDIF
#
# ---Print detail line
# line, 0 SAY $(NUMERO,1, 20)
# line, 23 SAY $(FIRMA,1, 40)
# line, 46 SAY $(TIPO,1, 15)
# line, 64 SAY $(GRAFICOS,1, 15)
# line, 82 SAY $(STN(ADICION, 2, 0),1, 2)
# line, 87 SAY $(STR(PUNTOS, 2, 2),1, 2)
STORE line+1 TO line
SKIP
```

```
Press RETURN to Continue
#NODD
# line+1,0 SAY " "
SET FORMAT TO SCREEN
RELEASE ALL
SET TALK ON
SET BELL ON
RETURN
# EOF: ARCH1.CMD
```

Para... PCW

type etl.cmd

```
# Program : ETI.CMD
# Author : Your Name
# Date : 00/00/00
# Notice : Copyright 1988, All Rights Reserved
# Reserved : select, condition, extra
#
SET TALK OFF
SET BELL OFF
STORE " " TO select
USE ARCH1
ERASE
# 2. 0 SAY "ARCHI MAILING LABELS"
# 2.72 SAY DATE()
# 3. 0 SAY "*****"
# 3.40 SAY "*****"
STORE " " TO select
# 5.0 SAY "Output to the screen or printer? (S/P) "
GET select PICTURE "I"
```

```
READ
DO CASE
CASE select = "S"
ERASE
CASE select = "P"
SET CONSOLE OFF
SET PRINT ON
OTHERWISE
ERASE
SET BELL ON
SET TALK ON
RETURN
ENDCASE
# ---Enter FOR (expression) for the labels, such as,
# ---STORE "STATE = 'CA'" TO condition
STORE " " TO condition
DO WHILE NOT EOF
IF condition <> " "
IF NOT (&condition)
SKIP
LOOP
ENDIF
ENDIF
STORE 0 TO extra
```

```
Press RETURN to Continue
? NUMERO
IF TRIM(FIRMA) = " "
STORE extra + 1 TO extra
ELSE
? FIRMA
ENDIF
? TIPO
? PUNTOS
?
DO WHILE extra > 0
STORE extra - 1 TO extra
ENDIF
SKIP
ENDDO
#
SET PRINT OFF
SET CONSOLE ON
? "THAT'S ALL FOLKS "
CLEAR
```

```
Press RETURN to Continue
SET TALK ON
SET BELL ON
RETURN
# EOF: ETI.CMD
```



LA ENSEÑANZA, EL LOGO Y LOS SEMICONDUCTORES

En los tiempos que corren se insiste mucho en aplicar las técnicas informáticas en el campo de la enseñanza. Existen experiencias en diferentes países, entre ellos el nuestro, y en algunos de ellos, hasta se van sacando conclusiones. En lo que sigue, analizaremos esta tendencia, a la vez que presentamos a algunos jóvenes estudiantes de electrónica un tema básico y conocido, desde una óptica distinta.





Existen algunos lenguajes de programación, llamados «de autor» que se emplean en el campo de lo que ha dado en llamarse ENSEÑANZA ASISTIDA POR ORDENADOR, y más recientemente ENSEÑANZA BASADA EN EL ORDENADOR. Su nombre procede de la circunstancia que se da, y que es el pretender que sea el propio profesor quien cree y elabore el programa de ordenador que apoyará la lección. En menos palabras, el profesor, es el AUTOR del programa que le servirá de soporte para su trabajo. Uno de estos lenguajes es el PILOT.

El lenguaje **LOGO** ha sido clasificado como lenguaje de ACTOR. Tal como algunos lo podríamos entender, las características didácticas de PILOT forman un subconjunto de las de **LOGO**. Quien tenga unas ideas elementales de los objetivos y filosofía de ambos, convendrá en afirmar que PILOT es más instrumental que **LOGO**. Este último es instrumental, y más cosas.

Sin pretenderlo, he iniciado algo que se va del tema, como es un comentario comparativo de dos lenguajes.

Es el estado actual de las técnicas creadas para apoyar la labor de las personas y organismos cuya misión es enseñar, cualquier lenguaje de programación es sólo un instrumento más.

Los jóvenes que ahora son estudiantes, tienen muy claro que los libros que están a su disposición son verdaderas maravillas en las que cualquier texto viene acompañado por fotografías de una calidad indudable, esquemas, diagramas, caricaturas, etc.

Además, en las aulas existe todo tipo de proyectores. En las casas están los televisores y los equipos de vídeo, que bien utilizados, son apéndices de lo disponible en el centro de enseñanza.

Revolución en las aulas

Llega el ordenador. No es caro. Es una novedad y se suma a las facilidades de que se dispone. Hay algún profesor que sabe algo de programación, y que todos los años desarrolla una lección que requiere un soporte gráfico y con animación.

Decide preparar un programa de ordenador para impartir el tema de los SEMICONDUCTORES. Lo prepara y sale algo como la siguiente parte de este trabajo.

Alguien, no hace demasiado tiempo, pensó en unos elementos químicos, extraordinariamente abundantes en la naturaleza, y por tanto, baratos. Pensó en la relación que podía existir entre su estructura atómica y sus características eléctricas. Aludimos al GERMANIO y al SILICIO. Haremos referencia exclusivamente al SILICIO. Hay razones para que su uso esté más generalizado, y la teoría, al nivel que la desarrollaremos es común a ambos.

El átomo de silicio, tal como se representa esquemáticamente en nuestra pantalla, tiene cuatro electrones en su órbita externa. Un átomo de estas características, necesita de otros átomos para formar la estructura cristalina del silicio, que a la postre, es la que podemos ver y tocar, y fundamentalmente, servirnos de ellas.

Sabido es, que átomos como éstos necesitan disponer de ocho electrones en su órbita externa. Como sabemos, dispone de cuatro.

Cuando uno de estos átomos está rodeado de otros cuatro iguales, tiene sus cuatro electrones, y comparte un electrón de cada uno de los átomos que le rodean. Así queda conformado con ocho electrones. La «necesidad» queda cubierta. Los libros que tratan el tema, presentan un dibujo que aclara lo expuesto.

Un caso concreto

Nuestro programa presenta una pantalla que tarda un tiempo determinado en completarse. Es un llenado progresivo que permite observar cómo cada átomo se va rodeando paulatinamente de otros, de tal forma que al término de la ejecución, todos quedan cubiertos con sus ocho electrones. Las facultades de repetitividad y recursividad de **LOGO** han hecho factible este efecto. Bien es cierto que la estructura dibujada es planaria y la realidad es espacial. Los métodos convencionales habituales, a lo sumo presentan perspectivas y/o dibujos con sensación de relieve.

Basándonos en esta pantalla, podríamos, mediante la adición de texto, pasar a tratar de la INMOVILIDAD DE LOS ELECTRONES. Lo que saben un poco de química, son conocedores de que el tipo de enlace atómico que nos ocupa, es llamado ENLACE COVALENTE. Ello implica, si bien no necesariamente, mala conductividad eléctrica.

La presencia de un campo eléctrico, junto a la circunstancia de la existencia de cristales de silicio a temperaturas muy superiores al CERO ABSOLUTO, como es la temperatura de trabajo de las personas y las cosas creadas por las personas, pueden hacer que algunos electrones no se mantengan tan inmóviles como en principio se podría presumir. En efecto, esto ocurre. Por ello, un cristal formado exclusivamente por átomos de silicio dispone de electrones susceptibles de irse de sus respectivos

ANALOGO

núcleos. Evidentemente este cristal no es conductor, pero tampoco aislante.

Se le llama SEMICONDUCTOR INTRINSECO. La pantalla de nuestro ordenador quiere representar su estructura.

En el lugar donde estamos, tendríamos que hablar de unos entes en cuya existencia hemos de creer. Son los HUECOS. Si un electrón se va de su núcleo el átomo neutro en principio se encuentra con el HUECO que dicho electrón dejara. En los estudios que se hacen a este nivel, se aconseja tratar este hueco, como si de una carga POSITIVA se tratara. Es lógico. Este hueco será ocupado por otro electrón de los menos inmóviles. En nuestro programa **LOGO** no hemos considerado estos aspectos del SEMICONDUCTOR INTRINSECO. Nos referimos a aquéllos que lo diferencian del AISLANTE PERFECTO.

Realmente, lo que nos interesa, es entrar en el estudio de la estructura del material con el que se fabrican los famosos transistores, los famosos circuitos integrados, los diodos, etc.

Pues bien, lo que los problemas no podrán esperar, es lo que a continuación les contamos.

Un cristal de silicio, puro como tal, no presenta características eléctricas interesantes.

Si mediante técnicas especiales, que no vienen al caso, conseguimos introducir en él, una cantidad controlada de átomos de un elemento químico diferente, habremos impurificado el «puro silicio». Las consecuencias de ello son extremadamente interesantes.

Supongamos que el «intruso» es el elemento químico llamado ANTIMONIO, cuyo símbolo químico es «Sb». Dicho elemento tiene en su órbita externa CINCO ELECTRONES. En cantidades pequeñas quedará enquistado en la estructura del silicio, de tal forma que aparece rodeado de cuatro átomos de este elemento. Estos cuatro átomos que rodean al átomo de antimonio (la impureza), comparten cada uno de ellos un electrón con él. De esta forma un electrón del átomo de antimonio, sobra. Dicho de otra forma, permanece desligado de la estructura. Diríamos que se trata de un electrón errante, un electrón con mucha movilidad. La energía necesaria para arrancar a este electrón de su núcleo, es pequeña. Cada átomo de Antimonio que se introduce como impureza, aporta un electrón susceptible de circular. Hemos llegado a lo que se llama SEMICONDUCTOR EXTRINSECO TIPO N.

El apoyo gráfico dado a esta parte del tema, mediante el programa **LOGO**, consiste en una pantalla en la que se representa el cristal de silicio (una pantalla entera de átomos). Se han borrado dos átomos de este elemen-

ta, y en la zona que ha quedado libre, se han intraducido dos átomos de antimonio. Previamente a esta pantalla se ha previsto la presentación en otro, de un átomo de la impureza pentavalente. En ambas se observa claramente la situación del quinto electrón.

El logo da soporte gráfico fácil a cualquier tema

En el programa se ha diseñado igualmente otra pantalla en la que aparecen ambos átomos de antimonio en la misma posición que ocupaban cuando estaban acompañados de los de silicio. Se pretende esquematizar o un más la representación. Cuando se trata gráficamente la unión P-N, los átomos de silicio ya no aparecen en ningún tratado. A partir de este paso, es preciso dirigir la atención del lector a otras cuestiones, tal como se verá.

El relato de las impurezas se repite para la presentación del «**SEMICONDUCTOR EXTRINSECO TIPO P**».

En este caso las impurezas están formadas por átomos en cuya órbita externa aparecen solamente tres electrones de valencia. Es el caso del elemento llamado INDIO, símbolo químico «In».

Al introducir un átomo de indio, éste quedará rodeado de cuatro átomos de silicio. Cada uno de ellos tratará de compartir un electrón del átomo de indio. Una de aquéllos se quedará con los ganos, ya que como hemos anunciado, el átomo de indio sólo puede compartir los electrones que tiene, tres.

La introducción de impurezas de indio, va a suponer la aparición de un HUECO por cada átomo introducido. Como tal HUECO nos hemos permitido la libertad de representarlo. Efectivamente, no se ve nada. Pero se ve el hueco.

En ambas pantallas en las que aparecen impurezas con el cristal de silicio, éstas se muestran con una tonalidad diferente. Detalle no fácil de percibir por el lector, en las fotografías.

La movilidad de los HUECOS (*supuestas cargas positivas*) puede comprenderse si se entendió el razonamiento sobre la movilidad de los electrones. Los HUECOS dejarán de serlo en el preciso momento en que a ellos acudan los electrones.

En el programa **LOGO** aparece una secuencia que simula un electrón moviéndose erráticamente por la pantalla. Esto no se puede apreciar en las fotografías, por lo que no se ha intentado.

Un átomo de antimonio al que se le escapa un electrón, se transforma en un ión de antimonio. Este tiene carácter electropositivo.

El programa **LOGO** presenta una pantalla en la que se muestra una pastilla de **SEMI-**

CONDUCTOR EXTRINSECO TIPO N. Se observan unas cuadradas grandes con un gran signo «+». Representan iones de antimonio.

Al lado de cada uno, en su parte superior, aparecen otros cuadrados más pequeños con un signo «-». Son los electrones asociados a cada ión. Electrones libres, no los olvidemos.

La denominación «**TIPO N**» procede de la particularidad que tiene de que sus cargas libres, llamadas también PORTADORES MAYORITARIOS, son electrones, y por tanto electronegativas.

Hay otra pantalla en el programa que quiere representar una pastilla de **SEMICONDUCTOR EXTRINSECO TIPO P**.

Un átomo de indio puede considerarse formado por un HUECO (*positivo*) y un ión de indio (*negativo*). Así se muestra en la pantalla **LOGO**.

Los cuadrados grandes representan iones de indio, negativos y los pequeños, HUECOS positivos.

La denominación TIPO P se deriva del hecho de que las cargas libres en este caso, son huecos electropositivos.

La unión de un SEMICONDUCTOR N y un SEMICONDUCTOR P que también se representa en nuestra programa, mediante otra pantalla, es objeto de lo que podría llamarse la última etapa de nuestra lección.

Mediante técnicas adecuadas podemos realzar la unión física de estos dos tipos de semiconductores extrínsecos.

Lo que ocurre en virtud de esta unión, es lo siguiente. En la zona próxima a la misma, se produce un fenómeno llamado de «DIFUSIÓN». Consiste en que los electrones del cristal N, se sienten atraídos por los huecos, también próximos del cristal P. A causa de ello se origina una recombinación de ambos tipos de cargas (son de diferentes signos). Estos electrones y huecos próximos a la zona de unión, desaparecen como tales cargas.

Este fenómeno ha sido representado en nuestro programa, quizá no muy adecuadamente. Se puede mejorar. En el cristal N han desaparecido algunos electrones, por lo que quedará cargado POSITIVAMENTE. El cristal P experimenta la desaparición de los huecos. Estos son positivos, por ello este cristal queda dotado de carga NEGATIVA.

Por último, se pretende hacer patente otro fenómeno que daría fin a la **LECCION**, no al tema, que podrá ser objeto de tratamiento en otra circunstancia.

Nos referimos a la formación de la **BARRERA DE POTENCIAL**. La DIFUSIÓN continuará manifestándose en zonas más alejadas de la unión P-N. Sin embargo, hay algo que hace que esta «recombinación» se detenga; frene. En otra pantalla de nuestro programa **LOGO**, se puede observar este hecho.

Los HUECOS y ELECTRONES que desaparecieron en las proximidades de la unión de-

jan «al descubierto» IONES NEGATIVOS de INDIO en el semiconductor P e IONES POSITIVOS de ANTIMONIO en el semiconductor N, respectivamente.

De esta manera, los electrones libres del cristal N que en principio tenderán a recombinarse con huecos del cristal P, no podrán hacerlo. La razón está en que, tal como nuestra pantalla muestra, existen en el cristal P, y en las proximidades de la unión P-N, IONES NEGATIVOS que van a repeler a las citadas cargas libres negativas (electrones). La misma va a ocurrir con los IONES POSITIVOS del cristal N, respecto a los huecos del cristal P. En esta consiste la BARRERA DE POTENCIAL. Es un impedimento para la continuación de la DIFUSIÓN. También influir en el comportamiento del cristal P-N cuando éste sea conectado a un manantial de corriente continua. Aquí empezará ya otra historia.

Ante todo, nos ofrecemos para facilitar los programas de SOFTWARE relativos a la lección expuesta.

Faltan muchos detalles. Si este trabajo no hubiese sido enfocado para la publicación en nuestra revista y si como apoyo en un verdadero clase de electrónica, habríamos tenido que añadir texto en todas las pantallas. También hubiese sido necesario perfeccionar, dentro de la que el lenguaje permite, algunas efectos que pudieran simular animación.

Modificación y actualización de las lecciones

De aquí podríamos conectar con el principio de nuestra artículo. Toda mejora y/o perfeccionamiento del programa implicará una inversión de tiempo. Puede llegar el momento en que lo consideremos terminado. Lo presentamos ante los alumnos de un centro de enseñanza. Estudiamos sus reacciones y recogemos sus sugerencias. Vuelta a la modificación, a la ampliación.

Pasa el tiempo, y el mismo centro de enseñanza ha cambiado.

Intentamos poner de manifiesto nuestras dudas e inquietudes ante situaciones ajenas al autor del programa, que son cambiantes incluso dentro del periodo de gestión, ampliación y sucesivas adecuaciones de éste.

Si o requerimiento de un nuevo equipo docente se toma la decisión de introducir cambios, ¿quién se encargará?

En cuanto a nuestro lenguaje de programación, el **LOGO**, queda claro que es perfectamente adecuado para estos fines. Imaginemos que en nuestro programa hemos añadido una especie de MENU para que en cualquier momento podamos acceder a cualquier pantalla, en función de las necesidades del alumno.

Split Personalities

ZAFIRO
CHIP



Un nuevo rompecabezas de acción para todas las edades que te divertirá y sorprenderá. Por fin, he aquí todas tus personalidades favoritas (o no tan favoritas) que pueden aparecer y aparecer en la comodidad de tu propia casa. Entremécete por la forma que su cursor gigante desplaza a "sir Olive", los Royals, Pop Starta, políticos y otros muchos en pedazos y piezas alrededor de la pantalla a un ritmo que pone los pelos de punta.

Spekman
Comodore
Amstrad
Amstrad DSK



ZAFIRO SOFTWARE DIVISION
Paseo de la Castellana, 141. 28046 Madrid
Tel. 459 30 04. Tel. Barna. 209 33 65. Télex: 22690 ZAFIR E

Editado, fabricado y distribuido en España
bajo la garantía Zafiro. Todos los derechos
reservados.


```
to cuadrado
repeat 4 [fd 40 rt 90]
end
to transla1
pu fd 5 rt 90 fd pd
end
```

```
to letra.s
fd 15 lt 90 fd 15 lt 90 fd 15
rt 90 fd 15 rt 90 15
end
```

```
to transla2
pu rt 90 fd 30 lt 90 fd 12 lt 90 pd
end
```

```
to letra.i
fd 20 pu fd 3 pd fd 2 pu bk 24
rt 90 pd fd 3
end
```

```
to nucleo.si
cuadrado
letra.s
transla2
letra.ic
end
```

```
to transla3
pu rt 90 fd 5 lt 90 fd 5 lt 90
end
```

```
to electrones
repeat 4 [fd 30 rt 90 fd 5 pd fd
10 pu b
k 15 lt 90 fd 10 lt 90]
end
```

```
to atomo.si
setpc 2
pd nucleo.si
transla3
electrones
end
```

```
to transla4
rt 90 pu fd 20 lt 90
```

```
to filas if :y < -140 [stop]
repeat 12 [atomo.si transla4]
make "y :y - 60
setx -330 sety :y
filas
end
```

```
to preparacion
make "y 161
cs ht pu setx -330 sety :y
end
```

```
to pantasi
preparacion
filas
if :y < -140 [stop]
end
```

```
to nucleo
end
to transla2a
pu fd 5
end
```

```
to letra.b
pd rt 90 fd 30 lt 90 fd 10 lt 90
fd 15 l
t 90 fd 10
end
```

```
to transla3
pu rt 180 fd 15 rt 90 fd 20 rt 180
end
```

to nucleo.sb	to atomo.sb
cuadrado	setpc 1
transla1	nucleo.sb
letra.s	electrones
transla2a	transla4a
letra.b	electron5
transla3a	transla5a
end	end

```
to transla4a
pu fd 40 lt 90 fd 20 rt 90 fd 5
end
```

```
to electron5
pd fd 10
end
to transla5a
pu bk 15 rt 90 fd 20 rt 90 fd 40
rt 180
end
```

```
to bonucleo
bocuadrado
transla1
boese
transla2
boi
end
```

```
to bocuadrado
pe cuadrado
end
```

to boese	to boi
pe letra.s	pe letra.i
end	end

```
to boelec
repeat 4 [fd 30 rt 90 fd 5 pd pe
fd 10 p
u bk 15 lt 90 fd 10 lt 90]
end
```

```
to boatomo
bonucleo
transla3
boelec
end
```

```
to impureza
pu setpos [-210 103] pd boa-
tomo pu setpo
s [-210 103] pd atomo.sb
pu setpos [150 -71] pd boatomo
pub setpos
[150-71]pd atomo.sb
end
```

```
to nucleo.in
pd
cuadrado
letra.i
transla2b
letran
transla3b
end
```

```
to transla1b
pu rt 90 fd 10 lt 90 fd 5
end
```

```
to letrai
pd fd 30 rt 90 fd 1 rt 90 fd 30
lt 180
end
```

```
to transla2b
pu rt 90 fd 10 lt 90
end
```

```
to lettran
pd fd 20 bk 5 rt 75 fd 16 rt 105
fd 19
end
```

```
to transla3b
pu fd 5 lt 90 fd 5 lt 90
end
```

```
to electro.in
repeat 3 [fd 30 rt 90 fd 5 pd fd
10 pu b
k 15 lt 90 fd 10 lt 90]
fd 30 rt 90 fd 15 bk 15 lt 90 fd
10 lt 9
0
end
```

```
to atomo.in
setpc 1
nucleo.in
end
```

```
to impureza.in
pu setpos [-120 101] pd ato-
mo.in
pu setpos [150 -79] pd boatomo
pu setpos
[150 -79] pd atomo.in
end
```


to ion.sb
 transla4a pd pe fd 10 pu
 setpos [-170 103]
 transla4a pd pe fd 10
 end

to movilidad
 seth 0 pd fd 10 wait 15 pe bk 10
 pu
 end

to elecmovil
 setpos [-210 160] movilidad
 setpos [-10 -100] movilidad
 setpos [200 50] movilidad
 setpos [10 -10] movilidad
 setpos [-250 10] movilidad
 setpos [250 -100] movilidad
 end

to elecmovilr
 repeat 10 [elecmovil]
 end

to pastillan
 pu setpos [-160 -30]
 pd repeat 2 [fd 210 rt 90 fd 160
 rt 90]
 end

to ionan
 pd repeat 4 [fd 20 rt 90]
 fd 10 rt 90 fd 20 bk 10 lt 90 fd
 10 bk 2
 0 lt 90 fd 10 rt 90
 end

to cruz
 fd 10 rt 90 fd 20 lt 90 fd 10 lt 90
 fd 1
 0 lt 90 fd 20 rt 90 fd 10 rt 90
 end

to columna
 repeat 3 [pd ionan pu fd 50]
 ionan
 end

to posic
 pu fd 10 rt 90 fd 10 lt 90 pd
 end

to posici
 pu fd 10 rt 90 fd 10 lt 90 pd
 end

to columnasn
 repeat 3 [columna pu fd 20 rt 90
 fd 40 r
 t 90 fd 170 rt 180 pd]
 columna
 end

to sicolibre
 pu setpos [-140 10]
 end

to colibre
 repeat 3 [pd eleclibre pu fd 50]
 pd eleclibre
 end

to eleclibre
 repeat 4 [fd 10 rt 90]
 fd 5 rt 90 fd 10 bk 10 lt 90 bk 5
 end

to tocolibre
 repeat 3 [pd colibre pu rt 90 fd
 40 rt 9
 0 fd 150 rt 180]
 pd colibre
 end

to semicondn
 pastillan
 situacion
 columnasn
 sicolibre
 tocolibre
 end

to letra.ic
 fd 20 pu fd 4 pd fd 4 pu bk 28
 end

to boic
 pd pe fd 29 bk 29
 end

to transla3c
 pu bk 5 rt 90 fd 8 lt 90
 end

to pastillap
 pu setpos [160 -30]
 pd repeat 2 [fd 210 lt 90 fd 160
 lt 90]
 end

to ionind
 pd repeat 4 [fd 10 lt 90 fd 20 lt
 90]
 end

to situacol
 pu fd 10 lt 90 f 10 rt 90 pd
 end

to columnaind
 repeat 3 [pd ionind pu fd 50] pd
 ionind
 end

to todion
 repeat 3 [columnaind pu lt 90 fd
 30 lt 9
 0 fd 150 rt 90 fd 10 rt 90 pd]
 columnaind
 end

ANALOGO

PROCEDIMIENTOS

to situacion
 pu fd 10 rt 90 fd 10 lt 90 pd
 end

to hueco
 repeat 4 [fd 10 lt 90]
 fd 5 lt 90 fd 10 bk 5 rt 90 fd 5
 bk 10 r
 t 90 fd 5 lt 90
 end

to colhueco
 repeat 3 [pd hueco pu fd 50 pd]
 pd hueco
 end

to todohueco
 repeat 3 [colhueco pu lt 90 fd 20
 lt 90
 fd 150 rt 90 fd 20 rt 90 pd] col-
 hueco
 end

to situahuecos
 pu setpos [140 10]
 end

to unionpn
 semicondp
 semicondn
 end

to semicondp
 pastillap
 situacol
 todion
 situahuecos
 todohueco
 end

to bocolibre
 setpos [-20 10]
 repeat 3 [pd pe eleclibre pu fd
 50]
 pd pe eleclibre
 end

to borraacol
 setpos [20 10]
 repeat 3 [pd pe hueco pu fd 50]
 pd pe hueco
 end

to difusion
 pu bocolibre
 pu borraacol
 end

to barrera
 pu setpos [-40 -30]
 pd repeat 2 [fd 210 rt 90 fd 80
 rt 90]
 rt 90 fd 40 lt 90 fd 1 pd pe fd 208
 pu b
 k 210
 end

Intercambio programas para **Amstrad CPC-464**, mejor comerciales, poseo más de 150 títulos, en la ciudad de La Coruña. Llamar al 29 92 50 a horas de comer. Preguntar por Santi.

Vendo ordenador ATARI 800 XL con datacorder original, dos joysticks con mando a distancia y conexiones para periféricos. Todo a estrenar y con garantía. Además regalo curso de iniciación al BASIC y dos juegos Atari. Precio a convenir. Mi nombre es Antonio y vivo en C/ de Basilio Sala 15-4 en San Juan de Alicante.

Vendo 1.º CPC-464 con monitor en fósforo verde, en perfecto estado, como nuevo. Monitor recién revisado. Regalo muchos programas de juegos. 50.000 pts. 2.º Impresora IBM gráfica, 80 c.p.s. bidireccional lógica optimizada, 40 a 132 columnas. 4 modos gráficos. EPSON compatible (MX-80). 48 tipos de escritura diferentes, con enfatizado, replicado, sub- u superíndices, subrayado, etc. Muy poco uso. Comprada hace sólo seis meses. Si tienes un **Amstrad** regalo TASCOPY y TASPRINT. Regalo 500 hojas de papel continuo. 65.000 pts. discutibles. 3.º Los dos juntos: sólo 100.000 pts. Llamar al (91) 742 28 37 y preguntar por Claudio o Diego.

ERRATA EN EL ANALOGO DEL NUMERO 60

Los que siguieron el artículo de Logo del número anterior verían que en él se hace mención a dos procedimientos: iguales e iguales 2. Deberían encontrarse en las dos fotos que rezan con dicho titular. Por desgracia, el color del fondo de la foto (*negro*) y el del listado (*negro*), coinciden, con lo que hay que hacer un magno esfuerzo visual para poder leerlo. Aquí van nuevamente. Esta vez, sobre fondo... ¡blanco!, los dos listados y nuestras disculpas.

PROCEDIMIENTO UNO

```
to iguales :mod :inp
if (and (:mod = []) (:inp = [])) [op 1 = 1]
if (or (:mod = []) (:inp = [])) [op 1 = 0]
if ((first :mod) = (first :inp)) [op iguales (bf :mod) (bf :inp)]
if (first :mod) = "any [op iguales (bf :mod) (bf :inp)]
if (first :mod) = "many [op iguales2 :mod :inp] [op 1 = 0]
end
```

PROCEDIMIENTO DOS

```
to iguales2 :mod :inp
if (iguales :mod (bf :inp)) [op 1 = 1] [op iguales (bf :mod) (bf :inp)]
end
```

MERCA COMPUTER

Tienda n.º 1 en Amstrad

TAMBIEN COMPATIBLE PC

TODO A LOS MEJORES PRECIOS

464 CPC FN	50.999
6128 FV	72.999
8256	115.999

MAS IVA

Comandante Zorita, 13 (tienda)
Telf. 253 57 93. 28020 MADRID

SERMICRO

SERVICIO TECNICO Amstrad

Barcelona
(93) 307 13 91

Sevilla
(954) 52 56 24

Valencia
(96) 154 11 43

La Coruña
(981) 22 10 12

Bilbao
(94) 444 60 87

Zaragoza
(976) 34 57 22

Málaga
(952) 32 09 44

P. de Mallorca
(971) 23 07 37

SEDE CENTRAL:
Avda. Ramón y Cajal, 107
Tel. 416 80 85

28043 MADRID

Tu séptima pieza y tu séptimo número

Recorta
y pega
esta
pieza
en su lugar.



7

Recorta este número y guárdalo hasta que tengas los restantes, después pégalo en su lugar correspondiente, de forma que las sumas horizontales y verticales coincidan (15).

MAS POSIBILIDADES PARA TU AMSTRAD



SILICON DISK

Es el sistema de almacenamiento externo más rápido del mercado. Realiza las mismas funciones que un disco convencional a una velocidad 20 veces superior. Emula una unidad 8 en caso de 2 discos como una tercera unidad. Ideal para trabajar con ficheros, hojas de cálculo, contabilidad, etc. Tiene una capacidad de 256K, conectando la ampliación de memoria de 256K se obtienen en 576K (en el CPC 464) y 630K (en el CPC 6123) muy superior a los PCW 8256 y 512.



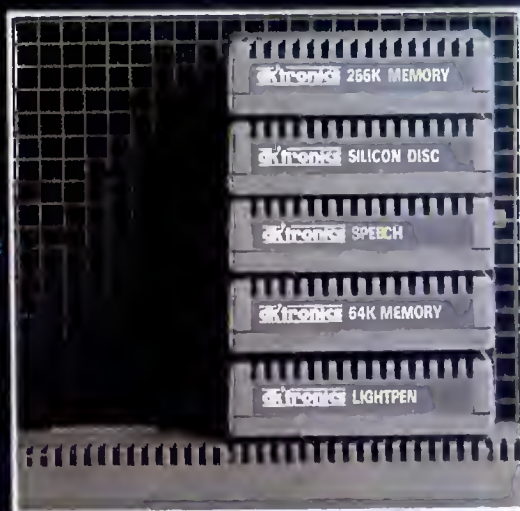
INTERFACE PROGRAMABLE CON SONIDO

Tiene las mismas posibilidades que el anterior y además incorpora el conocido chip de sonido AY-3-8912 que dispone la gama CPC. Con este periférico se igualan las prestaciones de sonido de los CPC 464 y 6123, disponiendo también de un altavoz externo que se incluye en el Kit.

AMPLIACION DE MEMORIA

La capacidad de memoria RAM de tu Amstrad puede ser aumentada. Dispone de una ampliación de 64K que convierte la memoria de tu ordenador, en la misma que monta el CPC 6123. Claro que si no tienes suficiente con 128K de RAM, existe otra interfase de 256K que aumenta la capacidad de tu ordenador a 320K de RAM, en el caso del CPC 464 y 380 en el 6128.

Todos los programas en CPM y CPM+ por fin a tu alcance.



VISITENOS EN EL SIMO 86
STAND G-97. PABELLON 9

MODULO HORARIO DE TIEMPO REAL

Este discreto módulo tiene las siguientes características:

Permite trabajar en tiempo real. Programable el día, hora, minutos y segundos, calendario, mes y año. Dispone de alarma programable y puede ser utilizado para su inclusión automática de discos y cualquier otra aplicación que requiera controles periódicos.

Todos estos periféricos se acoplan al port de salida posterior sin necesidad de desmontar el aparato.

NUEVO

INTERFACE PROGRAMABLE PARA JOYSTICK

Permite la posibilidad de conectar un joystick a tu PCW 8256/512. Son totalmente programables los movimientos arriba, abajo, izquierda, derecha y disparo.



SOFTWARE y PERIFERICOS
Importador en exclusiva de los productos

dktronics

COMERCIAL HERNAO, S.A.

Serrano, 30, 3.
Tel.: (91) 435 67 64 (4 líneas)
Telex: 47340 NAO-E
28001 MADRID

Deseo recibir información sobre

NOMBRE

DIRECCION

POBLACION

C.P.

PROVINCIA

CONVIERTE TU MONITOR EN UN TELEVISOR EN COLOR



AMSTRAD

C1M B40

12V DC

POWER

POWER
ON/OFF-10

ANTENNA

1-2-3-4-5-6-7-8-9-10

RECEIVER

DEALER

CHANNEL SELECT

■ AHORA PUEDES UTILIZAR TU MONITOR COMO PANTALLA DE ORDENADOR O COMO TELEVISOR ■

■ Empleando la más avanzada tecnología de los chips, el receptor DKTronics, convierte el monitor de tu CFC en un KIT de TV de alta calidad.



SOFTWARE y PERIFÉRICOS

Importador en exclusiva de los productos

dktronics

MÁS POTENCIA PARA TU AMSTRAD

■ COMERCIAL HERNAO, S.A. Serrano, 30, 3.º - Tel: (91) 435 67 64 (4 líneas) - Telex: 47340 NAO-E - 28001 MADRID ■